

SUMEX  
STANFORD UNIVERSITY  
MEDICAL EXPERIMENTAL COMPUTER RESOURCE  
RR - 00785

ANNUAL REPORT

Submitted to  
BIOTECHNOLOGY RESOURCES BRANCH  
NATIONAL INSTITUTES OF HEALTH

May, 1975

DEPARTMENT OF GENETICS  
STANFORD UNIVERSITY SCHOOL OF MEDICINE

## Table of Contents

Section	Page
List of Figures . . . . .	iv
I. RESOURCE IDENTIFICATION PAGE . . . . .	1
II. RESOURCE OPERATIONS . . . . .	2
II.A PROGRESS . . . . .	2
II.A.1 RESOURCE SUMMARY AND GOALS . . . . .	2
II.A.2 TECHNICAL PROGRESS . . . . .	3
II.A.2.a SYSTEM DEVELOPMENT AND OPERATIONS . . . . .	3
II.A.2.b USER SUPPORT AND INTERFACES . . . . .	15
II.A.3 RESOURCE MANAGEMENT . . . . .	18
II.A.4 FUTURE PLANS . . . . .	24
II.B SUMMARY OF RESOURCE USAGE . . . . .	27
II.B.1 RELATIVE SYSTEM LOADING BY COMMUNITY . . . . .	27
II.B.2 INDIVIDUAL PROJECT AND COMMUNITY USAGE . . . . .	30
II.B.3 NETWORK USAGE STATISTICS . . . . .	33
II.B.4 SYSTEM DIURNAL LOADING PROFILE . . . . .	35
II.C RESOURCE EQUIPMENT SUMMARY . . . . .	43
II.D PUBLICATIONS . . . . .	49
III. RESOURCE FINANCES . . . . .	50
III.A REFERENCE TO BUDGETARY DETAILS . . . . .	50
III.B PRELIMINARY SUMEX-AIM CPU AUGMENTATION PLAN . . . . .	51
III.C RESOURCE FUNDING . . . . .	53
IV. RESOURCE PROJECT DESCRIPTIONS . . . . .	54
IV.A FORMALLY APPROVED PROJECTS . . . . .	55
IV.A.1 STANFORD USERS . . . . .	55

IV.A.1.a	DENDRAL PROJECT . . . . .	55
IV.A.1.b	MYCIN PROJECT . . . . .	67
IV.A.1.c	PROTEIN STRUCTURE MODELING PROJECT . . . . .	72
IV.A.2	NATIONAL USERS . . . . .	79
IV.A.2.a	DIALOG PROJECT . . . . .	79
IV.A.2.b	DISTRIBUTED DATA BASES FOR CHRONIC DISEASES . . . . .	81
IV.A.2.c	HIGHER MENTAL FUNCTIONS MODELING . . . . .	83
IV.A.2.d	MEDICAL INFORMATION SYSTEMS LABORATORY . . . . .	84
IV.A.2.e	RUTGERS COMPUTERS IN BIOMEDICINE . . . . .	87
IV.B	INFORMAL PROJECTS . . . . .	94
IV.B.1	STANFORD PILOT PROJECTS . . . . .	94
IV.B.1.a	ARTIFICIAL INTELLIGENCE APPLICATIONS IN GENETICS . . . . .	94
IV.B.1.b	INFORMATION PROCESSING PSYCHOLOGY PROJECT . . . . .	96
IV.B.1.c	AIM RESEARCH - UNIVERSITY OF ROCHESTER . . . . .	99
IV.B.1.d	NATURAL LANGUAGE UNDERSTANDING . . . . .	104
IV.B.1.e	QUANTUM CHEM. INVEST. OF HEME PROTEINS AND FERREDOXINS . . . . .	105
IV.B.1.f	AUTOMATIC INTERMACHINE PROGRAM TRANSLATION . . . . .	108
IV.B.2	NATIONAL PILOT PROJECTS . . . . .	110

#### Appendix A

AI Overview by E. A. Feigenbaum . . . . .	115
---	-----

#### Appendix B

Justification for Storage Augmentation - July 1974 . . . . .	140
--	-----

## Appendix C

Assessment of System Responsiveness Under Load . . . .	149
--	-----

## Appendix D

PDP-11 SAIL Design Summary . . . . .	153
--------------------------------------	-----

## Appendix E

Subsystems and Documentation Directories . . . . .	158
--	-----

## Appendix F

Networking and Collaborative Research - DENDRAL Project . . . . .	167
--	-----

## Appendix G

Management Committee Membership . . . . .	194
---	-----

## Appendix H

User Information - General Brochure . . . . .	198
---	-----

## Appendix I

Detailed Questionnaire for Prospective New Users . . .	205
--	-----

## Appendix J

Response to Congressional Inquiry . . . . .	211
---	-----



## List of Figures

1.	Computer Configuration	. . . . .	111
2.	TYMNET Network Map	. . . . .	112
3.	ARPANET Network Map	. . . . .	113

NATIONAL INSTITUTES OF HEALTH  
DIVISION OF RESEARCH RESOURCES  
BIOTECHNOLOGY RESOURCES BRANCH

SECTION I - RESOURCE IDENTIFICATION

Report Period: Grant Number:  
From 1 August 1974 to 31 July 1975 RR-00785-02

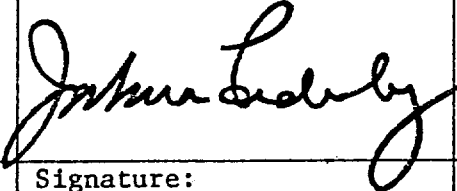
Name of Resource:  Stanford University Medical Experimental Computer	Resource Address:  Stanford University Stanford, California 94305	Resource Telephone Number:
Principal Investigator:  Joshua Lederberg, Ph.D.	Title:  Chairman and Professor Department of Genetics	Academic Department:  School of Medicine Department of Genetics
Grantee Institution:  Stanford University	Type of Institution:  Private University	Investigator's Telephone No.:  (415) 497-5801

Name of Institution's Biotechnology Resource Advisory Committee:

SUMEX-AIM Executive Committee

Membership of Biotechnology Resource Advisory Committee:

<u>Name</u>	<u>Title</u>	<u>Department</u>	<u>Institution</u>
Saul Amarel, Ph.D.	Chairman and Professor	Computer Science	Rutgers University
Donald Lindberg, M.D.	Professor Director	Pathology Information Science Group	University of Missouri School of Medicine

Principal Investigator:  Joshua Lederberg, Ph.D. Chairman and Professor	Signature: 	Date:  May 27, 1975
Stanford University Official:	Signature:	Date:

## II RESOURCE OPERATIONS

### II.A PROGRESS

#### II.A.1 RESOURCE SUMMARY AND GOALS

The SUMEX (Stanford University Medical EXperimental computer) project is a computer resource funded by the Biotechnology Resources Branch of the National Institutes of Health and encompasses a dual mission: 1) the promotion of applications of artificial intelligence (AI) computer science research to biological and medical problems and 2) the demonstration of computer resource sharing within a national community of health research projects. The SUMEX resource resides administratively within the Genetics Department of the Stanford University Medical School and serves as a nucleus for a growing community of projects, both within and external to Stanford. SUMEX provides computing facilities specifically tuned to the needs of AI research and communication tools to facilitate inter- and intra-group contacts as well as trial dissemination of research products to medical users. The project also develops tools for and takes an active role in stimulating community relationships among collaborating projects and medical researchers.

User projects are separately funded and autonomous in their management and are selected for access to SUMEX on the basis of their scientific and medical merits as well as their commitment to the community goals of SUMEX (see Section II.A.3 on page 18). Currently active projects span a broad range of application areas such as clinical diagnostic consultation, molecular biochemistry, belief systems modeling, mental function modeling, and instrument data interpretation (see Section IV on page 54).

Artificial Intelligence is a branch of computer science which attempts to discern the underlying principles involved in the acquisition and utilization of knowledge in reasoning, deduction, and problem-solving activities. Two recent reviews give some perspective on the current state of AI (see Nilsson, N.J., "ARTIFICIAL INTELLIGENCE", Information Processing 74, North-Holland Pub. Co. and Feigenbaum, E.A., extracts from an informal report to ARPA-IPTO, attached as Appendix A, page 115). Each authorized project in the SUMEX community is concerned in some way with the application of these principles to medical problems. The tangible objective of this approach is the production of computer programs which, using formal and informal knowledge bases together with mechanized hypothesis formation and problem solving procedures, will be more general and effective consultative tools for the clinician and medical scientist. The exhaustive search potential of computerized hypothesis formation and knowledge base utilization, constrained where appropriate by heuristic rules or interactions with the user, has already begun to produce promising results in the areas such as chemical structure elucidation, diagnostic consultation, and mental function modeling. Needless to say, much is yet to be learned in the process of

fashioning a coherent scientific discipline out of the assemblage of personal intuitions, mathematical procedures, and emerging theoretical structure of the "analysis of analysis" and of problem solving.

Our community building role is based upon the current state of computer communications technology. While far from perfected, these new capabilities offer much needed additional freedom in defining collaborative linkages, both within a given research project and among them. Several of the active projects on SUMEX are based upon the collaboration of computer and medical scientists at geographically separate institutions; separate from each other as well as from the computer resource. Another major goal of the network experiment is to enable diverse projects to interact more directly and to facilitate selective demonstrations of available programs to physicians and medical students. Even in their current developing state, such communication facilities allow access to the rather specialized SUMEX computing environment and programs from a great many areas of the country (and to some extent in Europe) for potential new research projects and for research product dissemination and demonstration.

This past year has seen the SUMEX resource become fully operational; the initially designed hardware configuration is installed, both the ARPANET and TYMNET network connections are finally installed and working, and the menu of available user software is filling out. The resource was formally dedicated in mid-November 1974 with a one day symposium held at Stanford to describe and illustrate the objectives, capabilities, and opportunities of the resource.

Over this year the complement of projects has increased from the initial group of 5 to include 9 formal projects and a group of informal pilot efforts. Already a number of examples of the benefits of inter- and intra-group interactions have come to light. There have also been substantial efforts to introduce non-affiliated research people to a number of the programs which are far enough along in their development. The management committees which help direct the allocation and development of the resource are functioning and are actively pursuing the recruitment of additional significant projects and establishing necessary resource allocation policies.

## II.A.2 TECHNICAL PROGRESS

### II.A.2.a SYSTEM DEVELOPMENT AND OPERATIONS

#### DEC PDP-10 Hardware:

At the time of the last report, the initially designed configuration was almost complete, lacking principally the high speed swapping storage and the network connections (see "Communications" on page 9 below for a summary of network status). On an interim basis, swapping was being done from the moving head disks and remote communication was handled through several IN-WATS lines to the eastern portion of the country.

A fixed-head swapping system had been selected from Digital Development Corporation (DDC) to be interfaced to a DEC Special Systems controller because of a greater capacity and transfer speed for the same money compared to equipment offered by DEC. The DDC fixed head disks also offered the option to organize the storage by pages (1 page = 512 36-bit computer words) as is desirable for TENEX operation. The technology for the device had been demonstrated prior to ordering but DDC nevertheless experienced problems in producing the needed drive and after a 4 month delay agreed to substitute a system based on an older and more expensive technology. In order to meet the capacity specifications of the original device, two of the substitute devices were required and agreed to at no price increase. The delivery was made in two stages; an interim slow device in October 1974 and the 2 final devices in early January 1975. In a very difficult situation, DDC proved to be highly cooperative and responsible in meeting their obligations.

The performance of the system improved dramatically with the installation of the fixed head swapping device as was, of course, expected. Average overhead for the system (even in the then fairly lightly loaded state) dropped from over 30% to about 10%.

As we approached the second year and facility loading increased, we projected that two aspects of the system would become bottlenecks; these were on-line file space and swapping efficiency. These projections were based on observed file space consumption and tests of system overhead under a simulated large program environment (LISP) on the IMSSS KA-TENEX system. The details of the tests and the plan (reproduced in Appendix B) suggested that substantial efficiency improvements could be achieved by adding memory and fixed-head swapping storage. The added memory allows more runnable jobs to be in memory at a given time to use cycles otherwise lost in I/O waiting to load a runnable job. The added fixed head swapping avoids overflow to moving head devices which are much slower.

The AIM Executive Committee reviewed and accepted this proposal. We have implemented the file system and memory augmentations as being immediately needed (within the lead time of hardware delivery). A current hardware configuration diagram is shown in Figure 1 (see page 111). As the high speed swapping store has been brought up to initial design goals only as of January 1975, we have delayed the addition of more fixed head disk space pending demonstration that that is the most effective place to augment the system.

We have been observing system performance over the past few months as the user load has increased and find that the loading during prime shift frequently approaches saturation in terms of system responsiveness for interactive users. This is illustrated in the diurnal loading data (see page page 35) where it can be seen that the load average peaks typically run between 5 and 10 during prime hours(1). The reports from the individual projects (see Section

---

(1) The "load average" signifies the number of jobs waiting in queue to be processed at a given instant: it measures the number of people awaiting service at that moment, so that responsiveness will be

IV, particularly page 80 and page 93) in the SUMEX-AIM community verify that the loading is subjectively approaching saturation with statements expressing concern over the ability to work during prime time and to be able to have physicians use the interactive programs with enough responsiveness so that their frustration does not go so high as to discourage them from further use.

We have asked other users to gauge their experiences and those of their medical collaborators against load average measurements as well. Their additional comments, along with those in the individual project reports, are summarized in Appendix C, together with a discussion to relate these subjective assessments to objective system operation. Most users express difficulty about being too precise in their judgements but generally agree that very noticeable response degradations set in when the load average gets above about 4 or 5 and that responsiveness deteriorates increasingly (non-linearly) above that.

Of course, the loading is mitigated at non-prime time although a number of INTERLISP users (particularly students) work in the evening and night to get a less loaded machine. We are especially open to users who can effectively use the night hours (e.g., Hawaiian and Houston-UK cooperations. Also we are developing a batch capability to off-load some of the daytime work which is not interactive-critical. However, the very nature of interactive computing in consultative programs is that human beings are involved and the work commitments of professional people such as physicians, imply that their main load in using the machine will be during prime time. As the user community grows over the next year or so, these problems in achieving acceptable program interaction response will become all the more acute.

Our performance evaluation efforts have been to better identify the bottlenecks to system throughput and, within the council-approved funding levels, to judiciously implement augmentations which maximize efficient use of the resources for the community. Having implemented previously proposed memory augmentations, we are now observing quite low overhead times, generally from 10 - 20%. Based on recent observations of drum space usage (see page 42 showing a recent diurnal drum space loading plot) we find under heavy load, where subjective response time is unsatisfactory per user comment, that we are just at the point of exceeding the existing capacity. However, because users are not always diligent about freeing up allocated pages when idle (by RESETting), we may be able to make more effective use of the available space by system software to transfer dormant pages to the larger moving head disk(2). We have made some estimates of

---

(approximately) inversely related to the load average. Two, three, or even four times as many users may be connected to the system at such times; but users typically take time out to ponder what the computer has reported, or the jobs may be preoccupied with input or output rather than the CPU.

(2) This is a particularly striking example of the trade-off between hardware and software investment. The economy of a software solution is enhanced by the ease with which such system programs can be shared with other facilities.

effectiveness of drum utilization and find that a substantial number of pages (exact amount varies widely but typical estimates may be in excess of 20-30%) are really dormant and could be moved to moving head disk without degradation. This would free up these pages on the fixed head devices allowing more effective use.

We are then faced with the fact that SUMEX is becoming response-time limited during prime hours. An analysis of existing data and user comments given in Appendix C points to two aspects of the machine configuration contributing to the bottleneck; CPU capacity and memory. These resources are closely inter-coupled in the performance of a time-sharing system as pointed out in the appendix and must be balanced in a well tuned system. We are at a reasonable balance point for the present configuration but have run out of inherent capacity to support current and anticipated peak loads. The system operates efficiently at between 15 and 20% overhead but does not have the speed to complete pending jobs fast enough to ensure adequate interactive response time. Our judgement, based on the arguments in Appendix C, is that the highest priority augmentation at this time should be in CPU capacity to alleviate this problem.

We are actively investigating ways in which CPU capacity can be augmented to eliminate this bottleneck. A preliminary plan is given in the budget explanation for the next grant year which proposes upgrading the present KI-10 CPU to a KL-10 (See page 51). At this time, before KL-10 deliveries have even begun, there are a number of technical uncertainties in the plan which we are working to resolve. We feel it is very important to maintain a degree of flexibility in being able to respond to needed augmentations to eliminate such bottlenecks as the community grows and its needs become more clearly defined. Consistent with this view, we request that unobligated money from year 02 be carried forward so that when the CPU augmentation plan is refined and reviewed by the Executive Committee and BRB, this money may be used to provide the needed additional processing capability for the SUMEX-AIM community.

#### TENEX Software - Monitor:

SUMEX is running release 1.31 of the TENEX system with modifications to accommodate the KI-10 paging hardware. Paging on the KI-10 was introduced by DEC after BB&N's (Bolt, Beranek, and Newman) experience with PDP-10 paging using a BB&N-designed pager on the older KA-10. Unfortunately, DEC did not incorporate all of the page handling hardware features of the BB&N device which facilitate demand paging. As a result, some of the hardware features for dynamically determining which pages have been modified, for changing user context, and for specifying per page access status are missing and must be simulated in software. Whereas the KI-10 hardware is intrinsically about 2 times faster than the KA-10, this additional overhead reduces the effective speed ratio to between 1.5 and 1.8 depending on the load (under light load the higher figures are achievable). Over the past year, Mr. Rainer Schulz has made important improvements to the KI-TENEX paging software and scheduler logic to significantly reduce the software overhead under heavier loads thereby better approaching the higher speed ratio.

In addition to the KI-TENEX performance improvements, we have also written new sections of code to interface the high speed swapping devices, to accommodate the network interfaces (see "Communications" on page 9 below), to control to a first approximation the CPU allocation to users, and to allow alphameric account specifications for a more transparent scheme to account for facility use among the various projects and communities. The swapping storage handler offers several features including management of multiple devices, full use of the hardware command queuing features of the DEC special systems controller, and on-line diagnostic exercising.

The resource allocation scheme includes at this time primitive facilities to control the amount of CPU time an individual user can receive consistent with the system load. Each user is assigned a percentage which defines the absolute fraction of the machine he is nominally entitled to. As he exceeds this amount, his priority within the system is penalized more and more. The time scale for this adjustment is currently 90 seconds - after each such period, penalties are reset and competition begins anew. As a given user's priority decreases, other jobs will be run preferentially if possible. If there are no other runnable jobs, then the available time is allocated to users over their aliquot so as not to waste machine capacity.

We are continuing to investigate the appropriate policies for allocation control. In particular the use of absolute versus relative priorities implies that if no one achieves his allotment, then competition essentially reverts to a laissez faire system. Also, it is becoming clear that people with fixed personal schedules (program demonstrations or busy clinicians) require some sort of reservation capability so that they can use the machine with reasonable responsiveness when they can arrange time rather than when the machine is relatively lightly loaded. We will continue to investigate the best approach to this problem in conjunction with the policy views of the management committees.

There, of course, have been periodic bugs in the software (reliability data are detailed below under "Reliability", see page 12) occasioned for the most part by monitor development efforts (network installation, swapping storage installation, etc.) but also present to a much less degree in the basic code. Aside from design oversights in development activities, the other bugs have been caused primarily by incomplete argument checking within system functions and calls - the intended use of a given section of code is often extended by the energetic user revealing these problems. It is a truism that the best way to check a system out beyond the basic functional state is to let time-sharing users on - on the whole, TENEX has borne up very well under such stress.

Operationally we have put a number of aids in the system to assure less operator error in bringing the system up. These include making sure the communications interfaces are enabled, automatically setting the time of day from other machines on the ARPANET when possible, automatically setting preplanned system halts to give users a maximum warning, various device exercisers with human readable error logging and decoding, and continuous system load monitoring and recording for diagnostic and management information purposes.



## TENEX Software - Executive:

Another area of software development is in the Executive program which is the basic user interface to manipulate files, directories, and devices; control job and terminal parameter settings; observe job and system status; and execute public and private programs. As with all system work, we face a dilemma which is particularly strongly felt in this area; should we run a "standard" system or should we adapt things to user community needs and thereby tend to a "home-brew" system? This is a difficult issue in that in many respects the SUMEX community is special - it includes a broad spectrum of users from professional computer scientists and programmers to biomedical research scientists and clinicians. The latter group, of course, want a minimum impedance to using the performance programs they are interested in while the former group wants a rich assortment of system facilities and as much flexibility as possible. Since most systems are designed for the programmer community, we have adopted the viewpoint that controlled augmentations of the system must be made to accommodate the medical user. Much of this work is still in process and will be for some time. The key point of this effort is to introduce knowledge about the individual user into the system (such as his usual defaults in using system functions, his level of expertise coupled to on-line assistance, his domain of interest to alert him to new information and perhaps personalized system commands or macros convenient to his needs) so that he perceives a system tailored to his style and conventions in using the computer.

Within the existing staff we have made only initial progress toward defining our goals and implementation. We have a proposal pending with ARPA (in conjunction with members of the Computer Science Department and the Institute for Mathematical Studies in the Social Sciences) to augment the staff to concentrate more effort on this crucial interface. The name adopted is "intelligent terminal" project; in the long term with micro-computers coming of age, it is likely that a considerable amount of such individual user adaptation will reside in the user's terminal. This off-loads much overhead from central computing resources and places at the user's disposal uniform access to the range of resources tied together by networks - the "intelligent" terminal could have all of the detailed information about linking to various facilities available and ease the user's need to remember a variety of different protocols. At the same time it provides relatively uniform access to these resources, routine clerical tasks such as mail manipulation, calendar management and text editing could be handled.

We will continue to devote effort in this area in up-coming work. To date, we have made a few strides to allow the system to remember user subcommand selections through a session (such as in SYSTAT and LIST), to offer a user his own specified sequence of operations to be performed upon login (system status, reminders for today's activities, etc. - commands are stored in his own LOGIN.CMD file), and to add additional commands to the EXEC for user convenience. These include easy password modification and file preservation control (PURGE to allow combined DELETE and EXPUNGE on specific files and RETAIN to allow control over file version retention

and excess version disposition). Hardlined terminals are now automatically recognized so users do not have to specify type and this capability will be added where possible to network logins as well. The EDIT command has been modified to allow user selection of a preferred editor program (TECO, SOS, or TV) and to remember the filename and editor for future calls. We have also made the various status commands (JOBSTAT, USESTAT, and SYSTAT) more informative.

Another aspect of the EXEC we are working on is that of security and access control. We have diverse needs; regular users with valid access to all facility capabilities, guest users (principally physicians and scientists) who want to try out various performance programs applicable to their field, and other guests (primarily from other network facilities) interested in our system and software they may want to obtain. Within the file access and descriptor blocks of TENEX, we are setting up several classes of user: authorized users, guests, and network visitors. The guest facility is a simplified login procedure not requiring a name previously given to the system (we request name and affiliation data for our records and to ease future access) but requiring a password obtained from an authorized system or collaborator project staff member. The guest login will have access to a limited domain of programs - primarily message communication programs and working AI systems (e.g., chemical structure generator, MYCIN, glaucoma programs, PARRY, etc.). Guests will not be able to do general text editing, file manipulation, or program development.

The network visitor will have access only to specified files which are ready and approved for export. This implements our obligation to keep licensed software from being exported without vendor approval and at the same time offers reciprocity in software exchange which is a mainstay of the network community.

It should be pointed out that security systems are really only effective against relatively benevolent users. Many of the security schemes depend on the combinatorics of guessing passwords, and by writing clever and persistent programs can be circumvented. We have done what seems reasonable to prevent such occurrences both prospectively and by looking out for unusual activities in real time and retrospectively (The system now gives each user the most recent previous login time to help him spot possibly unauthorized use).

#### Communications:

A most crucial aspect of the SUMEX system is effective communication with remote users. From the user's viewpoint, the reality of using a remote computer as if it were next door depends almost singly on the ability to achieve the subjective feeling that a network connection is like a local telephone call to the computer. One way of achieving this goal is, of course, to hook up individual telephone lines for users at various places around the country. For the complement and geographical distribution of users contemplated in the SUMEX community, this would be prohibitively expensive (somewhere in excess of \$10,000 - \$15,000 per month based on early loading

expectations and sure to grow with time). In addition to these economic arguments for terminal access, networking offers other advantages for shared computing such as uniform user access to multiple machines and special purpose resources, convenient file transfers for software sharing and multiple machine use, more effective backup, co-processing between remote machines, and improved inter-user communications. We have therefore based our remote communication services on the existing networks - TYMNET and ARPANET - which allow foreign host access. These two networks complement each other; the TYMNET providing primarily terminal service with very broad geographical coverage and the ARPANET having more limited access but providing a broader range of communication services. Together, these networks give a good view of the current strengths and weaknesses of this approach.

Most of the experience to date has focussed on user terminal access to SUMEX-AIM via networks - primarily TYMNET. Our recent connection to the ARPANET has not been operational long enough to extensively assess its performance for this report although new cooperative relationships are in the process of being explored (e.g., remote file storage, processor backup, and multiple machine use). Also, for particular groups with especially convenient access to the ARPANET (Rutgers and Higher Mental Functions), work on SUMEX has been facilitated through the ARPANET connection.

Current network terminal facilities are not able to accomplish the illusion of a local call completely. Data loss is not a problem in network communications - in fact with the more extensive error checking schemes, data integrity is much higher than for a long distance phone link. On the other hand, networking has as its underlying principle that through shared community use of telephone lines, widespread geographical coverage is possible at substantially reduced cost. Our experience with individual telephone lines (IN-WATS), maintained for interim service until network facilities became operational, and network facilities bear out the cost advantages and attendant problems.

To operate 4 lines (at most 4 simultaneous users) to the east coast area cost up to \$6,000 per month including extra hour use fees. The corresponding network (TYMNET) charges are down by a factor of about 2-3 from that for a peak user load 1.5 - 2 times higher. The other side of the coin is that networks such as TYMNET are a complex interconnection of nodes and lines spanning the country (see Figure 2, page 112). The primary cause of delay in passing a message through the network is the time to transfer a message from node to node and the scheduling of this traffic over multiplexed lines. This latter effect only becomes important in heavily loaded situations; the former is always present. Clearly from the user viewpoint, the best situation is to have as few nodes as possible between him and the host - this means many interconnecting lines through the network and correspondingly higher costs for TYMSHARE, a profit-making company. Herein is the tension; to balance the unit cost of network operation against user acceptable response times. TENEX in some ways emphasizes this conflict more than other time-sharing systems because of the highly interactive nature of terminal handling (e.g., command and file

name recognition and non-printing program commands as in text editors or INTERLISP). We have connected SUMEX to the TYMNET in two places as shown in Figure 2 so as to allow more direct access from different parts of the country. Also local lines to more strategic terminal nodes are being considered for users in areas poorly served by the existing line layout.

The ARPANET, while designed for more general information transfer than purely terminal handling, has similar bottleneck problems in its topology (see Figure 3, page 113). These are reduced by the use of relatively higher speed interconnection lines (50 K baud instead of 2400 - 9600 baud lines in TYMNET) but response delays through many nodes become objectionable eventually as well. Such response problems have led to the installation of an IMP at SUMEX-AIM as related below.

We take very seriously the responsibility to provide effective communication capabilities to SUMEX-AIM users and will continuously look for ways to improve our existing facilities as well as investigate alternatives becoming available.

Technically speaking, the TYMNET connection was installed in late August 1974 and the hardware and software debugged during September. We began offering "routine" service during October and TYMNET use has increased steadily since then (see "Summary of Resource Usage", page 33). The interface is built around a Varian 620-L mini-computer supplied by TYMSHARE, Inc. with software to communicate with the other nodes in the network. Connection to the SUMEX KI-10 is through a direct memory port with an MX-10 multiplexor. The memory access improves character handling efficiency as far as the PDP-10 is concerned and allows aggregate high speed communication without excessive I/O bus loading. The TENEX software support to handle the shared ring buffers and to manage protocols between the PDP-10 and the TYMNET was developed by Mr Michael Heathman of the SUMEX project. This effort required several man-months. We have recently improved the measurement tools available to assess network responsiveness. As noted above (Figure 2), we have 2 4800 baud connections into the TYMNET to gain more direct access to the major trunks running from San Francisco to the Washington D.C. and New York areas.

We have had more than the expected share of hardware difficulties with the TYMNET interface. These have arisen primarily because the backplane wiring of the interface as supplied by TYMSHARE was too tight causing wire insulation to break around sharp bends at wire-wrap pins and causing logic element and power supply short circuits. These problems have gradually subsided as the most troublesome wires have been replaced as problems come up.

The ARPANET connection has been the subject of much administrative discussion within ARPA during the previous year and was resolved (so it appeared then) at the start of our second grant year by ARPA giving us permission to connect as a very distant host (VDH) to one of the other IMP's on the network. Whereas this was clearly suboptimal from many technical points of view, a VDH connection was much better than no connection so we proceeded to order the necessary

hardware and to design the software changes to TENEX. The VDH connection of a PDP-10 TENEX system had never been done before (other VDH's are generally small machines such as PDP-11's). This required a special interface design by BB&N and extensive interrupt level coding changes to the TENEX system to accommodate the VDH (approximately 4 man-months of work). The hardware and communication lines were installed by early January and debugging extended until late February. The time-critical hardware/software interactions required for the VDH protocols caused numerous problems in achieving a working design and produced a substantial overhead in the KI-10 when finally running. The VDH interface was finally operational in early March.

At that time, the combination of the increased load on the network IMP to which we were connected (coincidentally also located at TYMSHARE) together with the already slow response the ARPA office was experiencing in doing their computing on the OFFICE-1 computer (also on the TYMSHARE IMP), encouraged ARPA to reconsider the placement of an IMP at SUMEX. We pointed out (see Figure 3) that an IMP at SUMEX connected to the TYMSHARE IMP and also connected to the Stanford AI Lab IMP, would eliminate 4 of the 14 nodes between ARPA in Washington and the OFFICE-1 machine. ARPA agreed to this plan and supplied us with an IMP on which we have been operational as a local host since middle April. Because of the way the KI-10 interface was designed for the VDH connection (included a modified local host interface as a subelement), we were able to adapt the interface to be a local host with about 30 wiring changes and no additional cost. The change between being a VDH and operating as a local host with an IMP was dramatic. What were previously very sluggish communications, even between SUMEX-AIM and other hosts in the area (e.g., SRI and Stanford AI Lab), improved by a factor of from 3 to 5 in responsiveness and speed. We are still in the process of arranging for the installation of the additional line to the Stanford AI Lab which should be done by mid-summer.

We are being somewhat restrictive about the use of the ARPANET at the present time because of the developing policy position for the administration of the network. The administration will pass from the ARPA Information Processing Techniques Office to the Defense Communications Agency as of July 1975. At that time we expect new policies to be announced relating to access authorization and network usage cost allocation. Until these issues become clarified, we have protected the facilities for calling from SUMEX out to other sites on the ARPANET, allowing only those users who are affiliated with on-going ARPA contracts to use the facility. This also protects the SUMEX-AIM machine from acting as an expensive terminal handler for other machines - this function is better fulfilled by dedicated terminal handling machines (TIPS). All other facilities of the network connection (calling into SUMEX from anywhere on the ARPANET and FTPing files in or out of SUMEX) are available to anyone possessing an authorized directory and password for the SUMEX machine.

#### Reliability and Backup:

System reliability has been somewhat variable over the past

year; excellent under stable hardware and software conditions and degrading during monitor development periods (network interfacing, swapping storage installation, etc.) and during periods of hardware problems. The pertinent data are given below with indications of eras during which development took place.

SUMEX-AIM CRASH FREQUENCY (crashes/month)  
AND DOWN-TIME DATA (hours/month)

Crash Type	AUG	SEP	OCT	NOV	DEC	JAN	FEB	MAR
DEC HARDWARE	14	6	7	6	17	21	13	15
SOFTWARE	2	5	0	2	1	2	8	4
ENVIRONMENT	2	3	1	0	0	0	1	1
TYMNET HDWRE	0	9	8	5	5	0	7	0
UNKNOWN	0	3	0	0	0	0	1	1
DOWN-TIME								
SCHEDULED	105	78	69	43	87	130	161	134
UNSCHED	73	39	29	19	36	21	31	31

DEFINITIONS:

Crash = Any occasion on which an operational system must be restarted or reloaded. Multiple crashes while trying to reload are not counted unless the system comes up fully between crashes.

DEC Hardware Crash = Any crash caused by a failure in the PDP-10 hardware or peripheral equipment (CPU, disk, drum, etc.)

Software Crash = Any crash caused by a malfunction within the TENEX software system.

Environmental Crash = Any crash caused by power failure, air conditioning outage, lightning, etc.

TYMNET Hardware Crash = Any crash caused by the TYMNET hardware or the interface to the PDP-10. This includes only the times when a TYMNET problem causes the PDP-10 to crash and not the times when the TYMNET goes down and the PDP-10 continues in operation.

Unknown Crash = All other crashes in which the cause is not assignable.

Scheduled Down-time = Preventive maintenance time (6-8 hours/week), scheduled maintenance to repair non-critical component failures, and system development activities requiring a stand-alone machine.

Unscheduled Down-time = Time lost because of unexpected hardware or software failure. For the most part this is the time to diagnose and either repair the problem or to reconfigure the system and bring it up to run in a somewhat degraded mode until a later scheduled shutdown for permanent repair.

#### DEVELOPMENT ACTIVITIES AFFECTING RELIABILITY

Whenever development efforts are undertaken which affect the monitor, some period of unreliability may result causing more crashes than are representative of the overall reliability of the system. The following gives some insight into these development efforts as reflected in the above data.

Aug -> Sept/Oct 1974: TYMNET development and installation

Early Nov 1974 and Jan 1975: Drum code development and hardware installation.

Feb/Mar 1975: Installation and checkout of ARPANET VDH code

As can be seen, we have had periods of rather serious hardware unreliability stemming from highly intermittent problems. There were a series of infant failures in the DDC swapping devices requiring several head replacements and causing several severe file crashes. Also during periods when one or more of the swapping devices was down, swapping off of moving head disks reduced efficiency substantially. These problems appear to have been solved since April. Other components of the system which have given trouble are the TYMNET interface (already mentioned), the PDP-10 memories and the moving head disks. The KI-10 CPU has been very stable and given only one problem over the past year (an I/O bus driver).

Most of the hardware problems have been very hard to track down as they caused crashes perhaps once per day and would not recur under diagnostic testing (in general TENEX exercises system components harder than do diagnostics). DEC has been very responsive in helping to find the problems in contrast to last year - the problems have simply gotten harder. It appears that the troubles should settle down soon as a number of intermittent faulty components have been found at last. We consider it a first order of business to improve these statistics.

From the user's viewpoint, besides the obvious inconvenience of not being able to work during down time, the fragility of the highly interlinked TENEX file system has caused several occasions of having to backup to previous file system states. We save changed files daily and copy the entire file system to fresh disk packs weekly. Thus an unexpected crash may cause the loss of up to one day's worth of work - it in fact may take longer for a given user to reconstruct the lost work if complex debugging or development changes were involved and undocumented. When the system is known to be subject to intermittent crashes, we backup more often to protect users. We are also investigating other modes of backup, now that we are on the ARPANET, such as the Datacomputer at the Computer Corporation of America.

Our current schedule for system backup is early Sunday morning (Pacific Time). We do not have enough staff for around the clock coverage and while we have overlapped staff to provide some weekend support and have scheduled backup then, this down-time for backup has been inconvenient for some users. We have tried to be responsive to these demands in that file backups used to be done to magnetic tape (requiring up to 6 hours for our size file system). We replaced this procedure with direct disk pack copying to reduce the time to about 2.5 hours. This eases the burden in down-time for essential backup operations; the next step would be to have enough staff to allow backup during very early morning hours to inconvenience (at least some) users less.

Another aspect of reliability and backup is the need to assure computing service for critical demonstrations, lectures, and the like. We are attempting to establish such a relationship with existing TENEX sites locally who are on the ARPANET but a surprisingly large number of problems arise; administrative and technical. These include the mechanics of moving files around beforehand (if a machine is down, files cannot be moved after the fact), allocation of space and time on otherwise heavily loaded machines, software compatibility in terms of monitor and languages, and approval of arrangements through responsible funding agencies. We are still working on this type of backup with an immediate need to support the AIM Workshop at Rutgers this June.

## II.A.2.b USER SUPPORT AND INTERFACES

We have already addressed one aspect of user support from the system viewpoint; that of adapting system functions and defaults to individual users. The following are aspects of user support involving specific pieces of software made available and attempts to facilitate user contact with them.

### Languages and Utility Programs:

A great deal of work was done during the past year in bringing up and improving the menu of subsystems available to users. New languages include SITBOL (Stevens Institute PDP-10 SNOBOL), FORTRAN-10 (release 4A), SAIL (TENEX version), TBASIC (Dartmouth language



definition with debugger, subroutines, etc.), INTERLISP updates, MACRO-10 update, FAIL update, ILISP (UC Irvine version of LISP 1.6), BCPL-10/11 (brought up by Rovner at Rochester), BLISS-10/11, and a preliminary version of PDP-11 SAIL (by Mr. Clark Wilcox of SUMEX). The PDP-11 SAIL compiler implements a significant subset of the SAIL language in a compiler which is highly machine independent by design. Code can be generated at present for the PDP-10, PDP-11, and IBM 360/370. Other machines are under consideration. The compiler itself currently runs on either the PDP-10 or a PDP-11/45 (with 32K of memory). Additional design information is given in Appendix D. In conjunction with these have come a variety of new utility programs including LINK-10, and CREF.

Beyond the language-related additions are a range of programs for text editing (including a CRT-oriented text editor, TV by Mr. Pentti Kanerva of IMSSS), mail handling, text justification (PUB and some new macro libraries), budgetting, typescript recording, multiple job fork control, mathematical modeling (MLAB from Gary Knott at NIH), graphics (OMNIGRAPH by Mr. R. Sproull at Xerox PARC), and so on. Rather than try to enumerate all of the available programs here, a brief summary of the major subsystems available is in Appendix E.

In a number of cases, considerable difficulties arose in importing software from various sites. These problems came about for a variety of reasons including getting incomplete sets of source files, programs written to take advantage of special system features and conventions not adopted universally, and inherent differences between TENEX and TOPS-10 (see "Compatibility" below, page 17).

#### Documentation and Education:

A substantial effort was made to better document subsystems and bugs for the programs available at SUMEX. As we have imported much of the software from other sources, we use available documentation where possible, update and adapt it where feasible, and write or rewrite from scratch where necessary. The reader is referred to Appendix D for a current listing of the <DOC> directory containing the on-line documents and a summary of available hard copy documents. Dr. Nancy Smith has completely revised the SOS and PUB documents and Dr. Robert Smith of IMSSS has prepared a document describing the TENEX version of SAIL. In addition numerous user help documents have been prepared for initial system access, network use, and subsystem usage aids.

Courses were prepared and given at Stanford covering the system assembly language (MACRO-10) and the TENEX system calls (JSYS's) (Messrs. Heathman and Crossland); TENEX SAIL (Dr. Robert Smith); and a class will be given shortly on PUB which is a powerful text justification language (Dr. N. Smith).

In addition to these more or less formal user support efforts, a large fraction of available staff time goes to tracking down real or perceived bugs encountered by users working on the system. It may be an under estimate to say that in excess of 30% of staff time is allocated to this purpose. Through the LINK and SNDMSG facilities, the staff provides help to users wherever they are located.

### Compatibility Issues:

Over the past year, in our commitment to software importation where possible rather than reinvention, we have encountered the problems of software incompatibility between various machines and operating systems fairly often. This problem is present to some extent between various TENEX sites where different releases of the system or languages are run or where local system additions or changes create problems (some TOPS-10 systems or runtime programs are non-standard as well). It is felt most acutely, however, between TENEX and TOPS-10 systems. A number of the problems stem from operational issues; file names or locations may be "hardwired" into a program and the same convention does not apply at other sites or file search pathways and hierarchies are not identical at all sites. These problems may be quite baffling without source files or deep insight into the program. These kinds of problems point up examples of where improved programming practices would make exportation of software more easily managed.

More difficult problems arise over basic incompatibilities between the TENEX and TOPS-10 systems. These arise either because languages of the same name (ignoring modifiers such as "TENEX" and "DEC" which anticipate problems) are not really the same or because definitions and design features of the two systems are inherently different. Such problems have been particularly frustrating for some of the people and programs originating from the NIH-DCRT machines but may be present for any program designed to run on the TOPS-10 system.

One area of considerable effort by Dr. R. Smith over the past year has been in narrowing the differences between TOPS-10 SAIL and TENEX SAIL. He has implemented a number of default line editing options and pseudo-interrupt options (e.g., control O to terminate terminal output) so that these functions will be transparent between the two machines. Other areas are more difficult to deal with including file system structure and protection and system calls. The TENEX file system is more elaborate than the TOPS-10 system in a number of ways such as naming conventions and the accommodation of multiple versions of a file as well as procedures for getting rid of files. In other ways, such as protection, different conventions were adopted. Through proper choice of defaults within the TENEX directory specifications for a given user, the naming problem can be mitigated; however, long names will still be recognized by TENEX and not by TOPS-10. The protection differences cannot be completely fixed either as the mapping from one system to the other cannot be easily made in all cases.

The issue of system calls is another difficult area; in TENEX the system calls (JSYS's) are different than those in TOPS-10 (UUO's). They are implemented using the hardware, however, in non-interfering ways so that it is possible to write an emulator program (PA1050) in TENEX which traps all DEC-style calls and translates them into "equivalent" sequences of JSYS calls. The problems arise when a) there does not exist a functionally equivalent translation or b) DEC has created a new UUO (as continually happens with new TOPS-10 releases) which the emulator does not know about. Mr. J. Crossland

(and before him S. Reiss) has spent considerable effort in tracking down and repairing as consistently as possible these kinds of problems. A major difficulty is that the original PA1050 emulator was written for a much earlier DEC system and has grown and been updated in something of a "crazy quilt" fashion. We estimate 1 - 2 man-years of effort to properly redo the package.

Many of the compatibility problems can be avoided prospectively through proper programming practices. This does not alleviate the difficulties in adapting older programs - a conversion effort of some sort is necessary although once done, through conditional compilation statements (say in SAIL), future compatibility can be maintained while continuing development on only one copy of the source program. We will continue to work to minimize these headaches and remain available to advise and help users as much as possible. Despite these compatibility difficulties we feel that the choice of TENEX was the correct one for the AI mission of SUMEX-AIM, primarily because of the advantages of the demand paging LISP environment uniquely available in TENEX.

#### Library Building:

Another aspect of user community support and a key element in the community-oriented mandate of SUMEX-AIM is the assimilation of software tools from active groups within and without the immediate SUMEX user groups. We have begun an effort to accumulate useful SAIL library routines from the various groups which have been working with this language (Stanford AI, IMSSS, SRI, NIH, USC-ISI, etc.). It is somewhat surprising that so little communication of SAIL library programs has taken place - it is almost literally true that each user has his own stock of tools in private procedure libraries. We have sent a letter to interested groups soliciting inputs on a basis which attempts to balance the problem of assuring library quality and integrity against establishing so high a threshold for quality and polish that individuals are not motivated to cooperate. This effort has just recently begun and no results are reportable to date.

### II.A.3 RESOURCE MANAGEMENT

Over the past year, the SUMEX project has devoted a substantial part of its effort toward its community-building role in recruiting new project, promoting interactions between user projects, and encouraging dissemination of running performance programs to medical scientists. A representative summary of SUMEX's community orientation in outlook and in action is given in a paper on networking and collaborative research (see Appendix F). This paper will be presented at the 170th American Chemical Society symposium this August 1975 and will appear in the proceedings.

The following summarizes specific aspects of SUMEX-AIM community management activities.

#### Dedication:

The SUMEX resource, having reached fully operational status by early fall 1974, held a dedication program at Stanford University on November 14, 1974. The program was an all-day symposium with the morning devoted to technical presentations by the initially authorized projects (see Section IV: DENDRAL, RUTGERS, MYCIN, Higher Mental Functions Modeling, and Protein Structure Modeling). The afternoon session addressed more global policy issues related to resource sharing and included presentations by Dr. Lederberg (SUMEX Principal Investigator), Dr. Thomas Bowery (Director of the NIH Division of Research Resources), Dr. W. Miller (Provost of Stanford University), and Dr. J.G.R. Licklider (Director of ARPA's Information Processing Techniques Office).

In addition to the program at Stanford, the attendant press releases and handout brochure (Appendix H), we also published an announcement of the SUMEX resource in the September 1974 SIGART (Special Interest Group for Artificial Intelligence) newsletter of the Association of Computing Machinery (ACM) and made a series of presentations on SUMEX and its related projects at the SIGBIO session of the 1974 annual ACM conference in San Diego (November 13, 1974).

#### Management Committees:

The SUMEX-AIM resource is constituted to attempt to bring into closer contact collaborating health research groups from around the country. This mission entails both the recruitment of appropriate research projects interested in medical AI applications and the catalysis of interactions among these groups and the broader medical community. As this effort is not a unilateral undertaking by its very nature, we have created several management committees to assist in administering the various portions of the SUMEX resource. As defined in the SUMEX-AIM management plan adopted at the time the resource grant was awarded, the available facility capacity is allocated 40% to Stanford Medical School projects, 40% to national projects, and 20% to system development and related functions.

Within the Stanford aliquot, Dr. Lederberg has established an advisory committee to assist him in selecting and allocating resources among projects appropriate to the SUMEX mission. The current membership of this committee is listed in Appendix G.

For the national community, two committees serve complementary functions. An Executive Committee oversees the operations of the resource as related to national users and makes the final decisions on authorizing admission for projects. It also establishes policies for resource allocation and approves plans for resource development and augmentation within the national portion of SUMEX. The Executive Committee oversees the planning and implementation of the AIM Workshop series and assures coordination with other AIM activities as well. The workshops are being carried out under Dr. S. Amarel of the Rutgers Computers in Biomedicine resource. The current membership of the Executive committee is listed in Appendix G.

Under the Executive Committee functions an Advisory Group representing contact with medical and computer science research relevant to AIM goals. The Advisory Group serves several functions in advising the Executive Committee; 1) recruiting appropriate medical/computer science projects, 2) reviewing and recommending priorities for allocation of resource capacity to specific projects based on scientific quality and medical relevance, and 3) recommending policies and development goals for the resource. The current Advisory Group membership is given in Appendix G.

These committees are actively functioning in support of the resource. Meetings to date have been held by telephone conference for the most part owing to the size of the groups and the difficulties in arranging for travel to meet face to face. These "missings" (a term coined by Dr. Licklider), in conjunction with terminal access to related text materials, have served quite well in accomplishing the agenda business and facilitate greatly the arrangement of meetings. A few technical problems occasionally attend such sessions such as poor telephone reception for some members but in general this approach is quite satisfactory.

#### New Project Recruiting:

As a result of the public announcements of the SUMEX resource, NIH reviews of the Health Manpower Act (769-A) proposals, and personal contacts by the staff or committee members, a number of additional projects have been admitted to SUMEX; others are working tentatively as pilot projects or are under review. We have prepared a variety of materials for the new user ranging from general information such as is contained in the brochure (Appendix H) to more detailed information and guidelines for determining whether a user project is appropriate for the SUMEX-AIM resource. Dr. E. Levinthal has prepared a questionnaire to assist users seriously considering applying for access to SUMEX-AIM (see Appendix I). Pilot project categories have been established both within the Stanford and national aliquots of the facility capacity to assist and encourage projects just formulating possible AIM proposals pending a formal review.

The projects newly admitted over the past year include (see Section IV for more detailed descriptions):

#### Stanford - (Pilot)

- 1) Information Processing Psychology; Drs. E. Feigenbaum (Stanford) and H. Cohen (UC San Diego)

#### National -

- 1) Diagnostic Logic Project (DIALOG); Dr. H. Pople and J. Myers, M.D. (University of Pittsburgh)
- 2) Medical Information Systems Lab (MISL); Dr. B. McCormick and M. Goldberg, M.D. (University of Illinois at Chicago Circle)

- 3) Distributed Data Base System for Chronic Diseases; Drs. F. Kuo (University of Hawaii), R. Nordyke, M.D. (Pacific Health Research Institute), and Dr. C. Kulikowski (Rutgers University)

We believe, within the current system capacity and with proper scheduling and capacity allocation controls, that another 2 or 3 major projects can be accommodated plus 5 or 6 minor projects. Here major and minor magnitude refers only to amount of computer resource consumption and not to scientific quality. Clearly the admission of these additional projects is based upon the ability to direct system use to currently underloaded parts of the day. This may require management committee decisions making access for some projects conditional upon system use at non-peak periods as well as other measures to encourage load leveling.

For another perspective on the community of projects currently being supported by the resource, see Appendix J. This appendix contains material prepared in response to a congressional inquiry to NIH-BRB on the scope and cost of community support by the SUMEX resource.

As an additional aid to new projects or collaborators with existing projects, we have a limited amount of funds which are being used to support terminals and communications needs of users without access to such equipment. We are currently leasing 5 terminals and 3 modems for users and will be providing some foreign exchange lines to users to improve network response time.

#### Utility of Intergroup Coupling:

One of the central objectives of the SUMEX resource is to encourage routine contact between remote groups. This may manifest itself in a number of ways such as collaboration within a project between researchers who are not geographically close, interactions between research projects which are at separate institutions, and dissemination of research products to users not close to the necessary specialized facilities. We are developing examples of useful collaboration in all of these categories as is summarized in the individual project descriptions attached in Section IV.

Several of the approved projects already involve remote collaborations; Rutgers Computers in Biomedicine (between Rutgers University, Mt. Sinai Hospital in New York, Johns Hopkins University in Baltimore, and Washington University in St. Louis), Protein Structure Modeling (between Stanford University and UC San Diego), and Distributed Data Bases (between the University of Hawaii and Rutgers University). The following message quoted from the Protein Structure Modeling group points up the utility of network relationships for coordinating remote development activities:

Date: 2 JAN 1975 0010-PST

From: ENGELMORE

Subject: ADVANTAGES OF SUMEX FOR COLLABORATIVE RESEARCH

"Yesterday, I was engaged for several hours in a very interesting collaboration, involving SUMEX, Steve Freer and others at UCSD. I was debugging a program which Steve recently sent to me, and running into a variety of bewilderments. We then linked to each other, so that my program output came out on Steve's terminal as well as mine. He then would comment on the output, direct my attention to appropriate parts of the program, and suggest changes. We made remarkable progress in that mode; it was as efficient as having Steve and some of his colleagues sitting right next to me as I worked. Although I knew perfectly well that networks and links permit this mode of operation, actually doing it was a fascinating experience. For Freer, however, it was a revelation! He had no idea before this that two people, 500 miles apart, could both examine program output independently and simultaneously. It really turned him on.

Had I not been able to converse with the UCSD group in "real time", I very likely would have traveled to La Jolla and worked there. So I feel the system we have in SUMEX is a real time and energy saver."

In the second category we are also developing examples of mutually useful interactions between research groups. Because the programs are accessible through common communication services, remote interactive criticism and discussions are possible as the programs are being developed. The following note describing an interaction between the MYCIN group at Stanford and the DIALOG group at Pittsburgh illustrates the point:

Date: 14 MAR 1975 1903-PST  
 From: SHORTLIFFE  
 Subject: Demo Last Saturday

"Bruce Buchanan suggested that I tell you about a use of the SUMEX system that we experimented with last Saturday. Harry Pople's group at Pittsburgh was interested in getting some reaction to their DIALOG system, so we arranged a time last Saturday morning for a demonstration. Meanwhile, several members of the medical diagnostic group at Rutgers were also interested and asked to sit in. We therefore all linked to one another at a prearranged time, and for about 2 or 3 hours, Pople demonstrated their program and then watched while I ran it on a patient of my own choosing. A number of comments and questions arose which were easily handled by the link procedure, and when the demonstration was over we continued to discuss via the link a number of other topics of mutual interest including plans for the AIM conference at Rutgers in June. It was a very satisfactory way to 'meet' without burning up the long distance phone lines (they, of course, were all logged on via the TYMNET), and the incident

may therefore be of interest to you when you discuss some of the novel advantages of a national resource such as SUMEX."

Another form of intergroup collaboration is developing between the Rutgers project and the MISL project at Illinois. The Illinois group is planning to use the Rutgers glaucoma programs as an integral part of their research with the University of Illinois Eye Clinic. There have been a number of delays in getting this interaction working smoothly caused by the problems in getting network connections working, needed language support debugged, and finally getting the glaucoma programs to a state where routine access is possible for the Illinois project. These should be solved now and hopefully the next report will see an active collaboration between these groups.

Finally, those projects with programs beyond the early development stages (principally DENDRAL, MYCIN, Rutgers glaucoma, and Higher Mental Functions PARRY) have made substantial investments in liaison and programmer time to facilitate non-expert user interfaces to their performance programs. These resulting programs have then been made available to selected professionals outside of the development groups for experimental use and appraisal. In numerous cases, the network connections have allowed contacts with these users from areas quite remote from Stanford and where it would be impossible to mount the programs for lack of necessary specialized computing facilities. These contacts have produced promising results even at these early stages as described in the individual project summaries (Section IV). A major objective of the SUMEX project community is to continue establishing contacts with non-computer scientists in the various research areas under investigation and to demonstrate and evaluate the utility of the medical AI programs.

#### Resource Allocation Policies:

As the SUMEX facility becomes increasingly loaded, a number of diverse and conflicting demands can be identified which require controlled allocation of critical facility resources (file space and central processor time). We have already spelled out a policy for file space management; an allocation of file storage will be defined for each authorized project in conjunction with the management committees. This allocation can be divided among project members in any way desired by the individual principal investigators. No system allocation enforcement will be imposed as long as there is adequate file capacity left in the system to afford as much flexibility as possible to projects for temporary file space needs. However, when used space approaches system capacity, a variety of tools (verbal requests, deleted file expunging, and forced file archival) are available to ensure that projects observe their allocated space. So far the user community has been very cooperative and has responded to verbal requests for file space clean-up.

As described under "System Development Progress", we have



implemented a primitive CPU scheduling algorithm intended to ensure that no one user gets more than a fair share of the machine when other users are contending. As discussed there, it is likely that a more sophisticated scheme will be necessary to meet the community needs (fixed personal schedules and relative priority ratings). This may be implemented by some form of "reservation" system where some prescribed fraction of the machine can be expected for a given individual or project during a specified period at the expense of priority at other times. We are discussing these issues with the management committees to evolve the most beneficial policy for the SUMEX-AIM community.

As also mentioned earlier, we are developing a categorization of users in terms of access privileges. These range from fully authorized users to guests and network visitors in descending order of system capabilities. We want to encourage bona fide medical people to experiment with the various programs available with a minimum of red tape while not allowing unauthenticated users to bypass the advisory group screening procedures by coming on as guests. We will continue developing this mechanism in conjunction with management committee policy decisions.

#### AIM Workshop Support:

The Rutgers Computers in Biomedicine resource (under Dr. Saul Amarel) is actively working on plans for the first AIM workshop this June. The current plans call for a one day general session covering a range of topics related to artificial intelligence research, medical needs, and resource sharing policies within NIH. The following three days will include a more intimate set of working sessions to allow first hand experience with running programs for various prospective users and interested research people. The SUMEX facility will act as the computing base for the workshop demonstrations. We are in the process of working with Rutgers to provide backup modes for program demonstrations in the event of system failure.

#### II.A.4 FUTURE PLANS

##### System Performance:

In the next year we will work on improving system performance based on measurement data now being collected and evaluated. For example, we want to tune the working set size limits and logic to improve the trade-off between paging traffic and the number of jobs in core. We are working on implementing an algorithm to more efficiently utilize swapping storage by migrating dormant pages off to moving head storage. In parallel with our measurement efforts, other groups (USC-ISI) are debugging and testing TENEX systems with memories larger than 256K words.

These efforts will assist us to plan where key augmentations (memory, CPU, swapping storage, file access) could increase throughput as the AIM community grows. We are currently developing a plan to

overcome the CPU bottleneck which we feel will shortly become the most critically limited resource. Preliminary details of this plan are described under "Equipment" in next year's budget (see page 51). We will refine this plan and submit it to the Executive Committee and BRB for approval. We have requested that any unspent funds from year 02 be carried forward to year 03 for this purpose.

We will investigate bringing up version 1.33 of TENEX with necessary KI-10 modifications in order to stay current with other TENEX sites (and to facilitate maintaining an up-to-date INTERLISP subsystem) as well as evaluate the scheduler and resource allocation group features introduced by BB&N in 1.33 to see if they will assist the allocation controls contemplated for various user needs. To date KI-TENEX 1.33 is still being debugged at NASA-AMES and we will wait until the system runs smoothly and reliably before experimenting with it.

We plan to bring up a batch processing capability for those jobs which need not run interactively. A primitive system has been put together by USC-ISI and will be extended where needed (e.g., to allow multiple jobs, priority control so as not to compete with interactive work, etc.). In addition, we will add a hardcopy plotter (plotter available from another project) to the system along with a spooler to facilitate multiple use.

We will continue to refine the Executive program and capabilities for guest users.

We will also investigate ways of improving network communication services. This will include attempts to optimize our current facilities for users through better ties to the networks and selective lines to tie individual users into more advantageous access points. We will also continue to explore other network and communication alternatives as they become available over the next year. Specific goals include improved response times and increased output speeds. We expect the ARPANET link to improve about mid-summer with the addition of the other 50 K baud line to the Stanford AI Laboratory IMP.

#### Adaptive User Interfaces:

We plan to continue work toward a more adaptive system for users including both simplifying access for non-expert users and anticipating default parameter conventions of individual users. Longer term planning may look at more sophisticated user modeling and the possibility of putting such personalized interfaces into a user's terminal. We are now in the process of defining system calls which will make user information uniformly accessible to programs that choose to make use of it.

#### Software Facilities and Libraries:

There is a continuing need for improved documentation of various aspects of the system and of available programs. We will be up-

grading this material, particularly as it relates to the inexperienced user.

In general we are attempting to up-grade the various DEC-originated subsystems to the newest versions to increase the chance of compatibility. We have recently done this with FORTRAN and MACRO and will bring the other programs along as soon as possible. The whole issue of compatibility is one which will receive attention. We will not be able to commit the necessary resources within available manpower to redo the TOPS-10 emulator correctly, but will keep chipping away.

Some requests to look into additional software subsystems have been received and we will consider mounting them if the community develops a definite need. Suggested augmentations include the simulation language SIMULA, the medical information system language MUMPS, BPL or APL, and subroutine libraries such as the biomedical statistics package. We will continue our efforts in the building of user subroutine libraries, concentrating initially on SAIL.

#### Informal Information Access:

One characteristic of the SUMEX community is the diversity of information, formal and informal, which flows around the system or is available from users. We want to work on ways to capture that information and direct it to other interested individuals. We have two major repositories for information, the <DOC> directory which contains the formal documentation of system facilities and procedures and the <BULLETINS> directory which is an accumulation of informal and dynamic information. We will be working on capabilities both to ease the entry and cataloging of information into these areas and to assist in guiding the user to that subset which is of interest to him at a given time. These user-oriented lookup protocols are, of course, strongly related to the problems of adaptive user interfaces to the system and each will benefit from the experience of the other.

#### Community Management:

We will continue to work with the management committees to recruit the additional high quality projects which can be accommodated and to evolve resource allocation policies which appropriately reflect assigned priorities and project needs. We hope to make more generally available information about the various projects both inside and outside of the community and thereby to promote the kinds of exchanges exemplified earlier and made possible by network facilities. The first workshop will provide much useful information about the strengths and weaknesses of the performance programs both in terms of criticisms from other AI projects and in terms of the needs of practicing medical people. We plan to use this experience to guide the community building aspects of SUMEX-AIM.

## II.B SUMMARY OF RESOURCE USAGE

The following data give an overview of the resource usage from August 1974 (when we began keeping detailed statistics) through March 1975. There are four sub-sections containing data respectively for 1) resource usage by community (AIM, Stanford, and system), 2) resource usage by project, 3) Network usage data, and 4) measures of diurnal variations in system loading.

### II.B.1 RELATIVE SYSTEM LOADING BY COMMUNITY

The SUMEX resource is divided, for administrative purposes, into 3 major communities: user projects based at the Stanford Medical School, user projects based outside of Stanford (national AIM projects), and systems development efforts. As defined in the resource management plan approved by BRB at the start of the project, the available resource will be divided between these communities as follows:

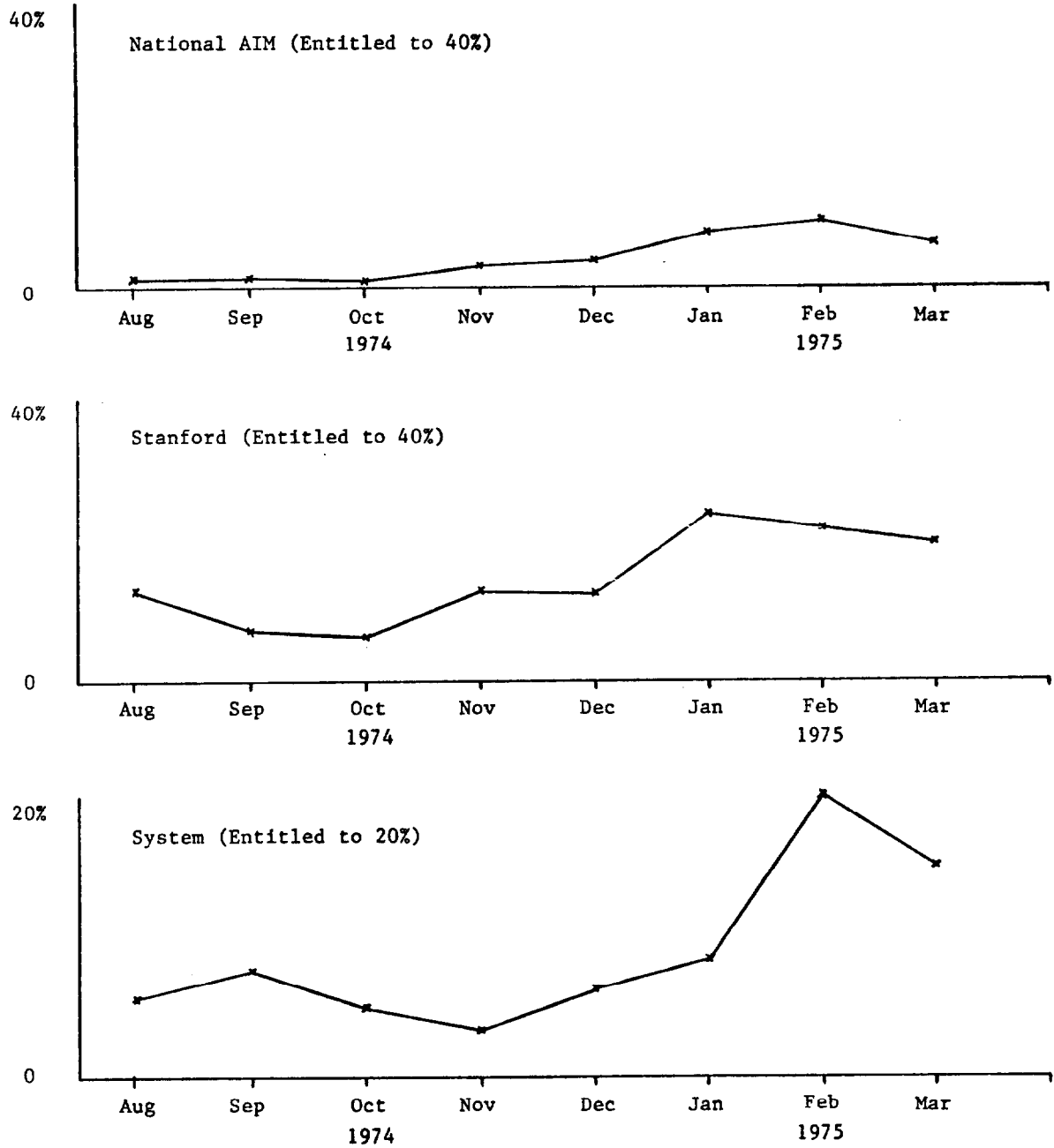
CPU Usage - Stanford	40%
AIM	40%
System	20%
File Space - Stanford	25,000 pages
AIM	25,000 pages
System	30,000 pages

We have recently brought an additional 40,000 pages of file space on-line (this happened since March and hence is not shown in these data) - this additional space will be equally divided, for the most part, between the Stanford and AIM user groups. The system file requirements include all of the subsystem files, documentation files, and other system-related files shared by all users in addition to staff directories. We expect the system allocation to grow somewhat more slowly than user space requirements, reflecting primarily the addition of new subsystems, documentation, and user information from time to time.

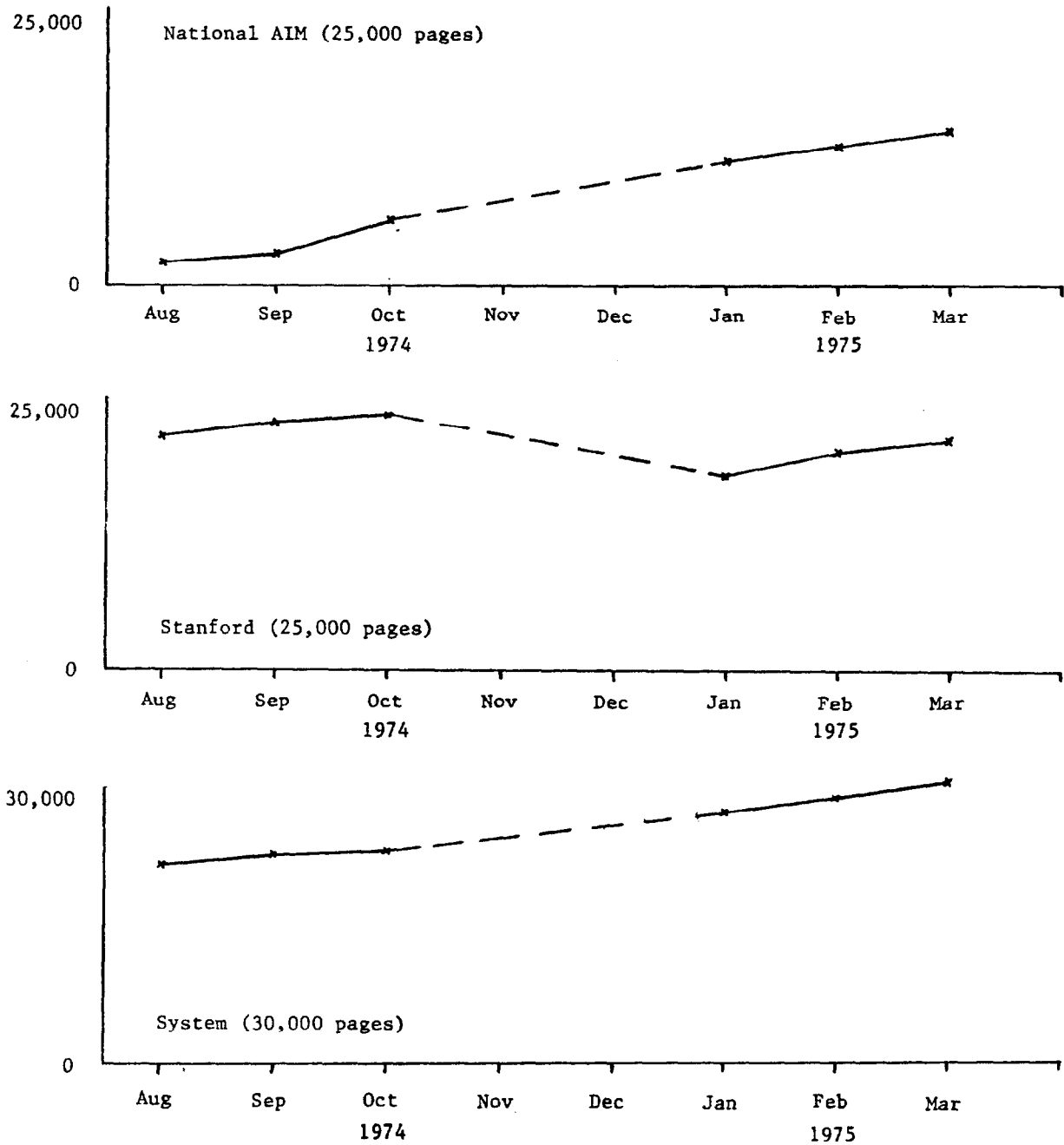
The following plots show monthly usage of CPU and file space resources for each of these three communities relative to their respective aliquots. Note that file space data are not available for November and December 1974 because we were making a transition between numeric and alphameric account designations.

As is evident in the data, many of the national AIM projects joined the SUMEX community during this year (as communication facilities became available) and their use of the facility began slowly. The rate of remote use is expected to increase substantially during the next few months.

## CENTRAL PROCESSOR USE BY COMMUNITY



## FILE SPACE USE BY COMMUNITY



## II.B.2 INDIVIDUAL PROJECT AND COMMUNITY USAGE

The table following shows average resource usage by project in the past grant year. The data displayed include a description of the operational funding sources (outside of SUMEX-supplied computing resources) for currently active projects, average monthly CPU consumption by project (Hours/month), average monthly terminal connect time by project (Hours/month), and average file space in use by project (Pages/month, 1 page = 512 computer words). Averages were computed for each project for the months since August 1974 or since the project was admitted to the resource if after August 1974.

Many of the national AIM projects joined the SUMEX community during this year (as communication facilities became available) and their use of the facility began slowly. For this reason, many of the average usage figures for these projects are lower than is representative of their activity currently. We expect the remote usage to increase significantly in the next months.

## RESOURCE USE BY INDIVIDUAL PROJECT

STANFORD COMMUNITY	CPU (Hrs/mo)	CONNECT (Hrs/mo)	FILE SPACE (Pages/mo)
1) DENDRAL PROJECT "Resource Related Research Computers and Chemistry" NIH RR-00612 (3 yr award) \$323,933 this year	46.38	728.5	12325
2) MYCIN PROJECT "Computer-based Consult. in Clin. Therapeutics" HEW HSO-1544 (3 yr award) \$124,000 this year	10.17	186.1	3264
3) PROTEIN STRUCT MODELING "Heuristic Comp. Applied to Prot. Crystallog." NSF DCR74-23461 (2 yrs.) \$88,436 total	3.62	139.0	933
4) PILOT PROJECTS (see reports in Sec IV.B.1)	6.49	208.4	4691
	-----	-----	-----
COMMUNITY TOTALS	66.66	1262.0	21213

## NATIONAL AIM COMMUNITY

1) DIALOG PROJECT "Computer Model of Diagnostic Logic" HEW MB-00144 (3 yrs.) \$125,027 this year	8.54	159.9	2063
2) Higher Mental Functions "Computer Models in Psychiatry and Psychother." NIH MH-06645 (3 yrs.) \$170,000 this year  NIH MH-27132 (2 yrs.) \$130,000 this year	0.12	11.2	1021
3) MISL PROJECT "Medical Information Systems Laboratory" HEW MB-00114 (2 yrs.) \$380,619 this year	0.68	53.1	242



4) RUTGERS PROJECT	7.26	351.5	2659
"Computers in Biomedicine"			
NIH RR-00643 (3 yrs.)			
\$285,240 this year			

5) AIM Administration	1.93	66.2	2587
	-----	-----	-----

COMMUNITY TOTALS	18.53	641.9	8572
------------------	-------	-------	------

## SUMEX SYSTEM

1) Development	36.97	1441.5	8450
----------------	-------	--------	------

2) Operations	6.04	618.9	17272
	-----	-----	-----

COMMUNITY TOTALS	43.01	2060.4	25722
------------------	-------	--------	-------

	=====	=====	=====
--	-------	-------	-------

RESOURCE TOTALS	128.20	3964.3	55507
-----------------	--------	--------	-------

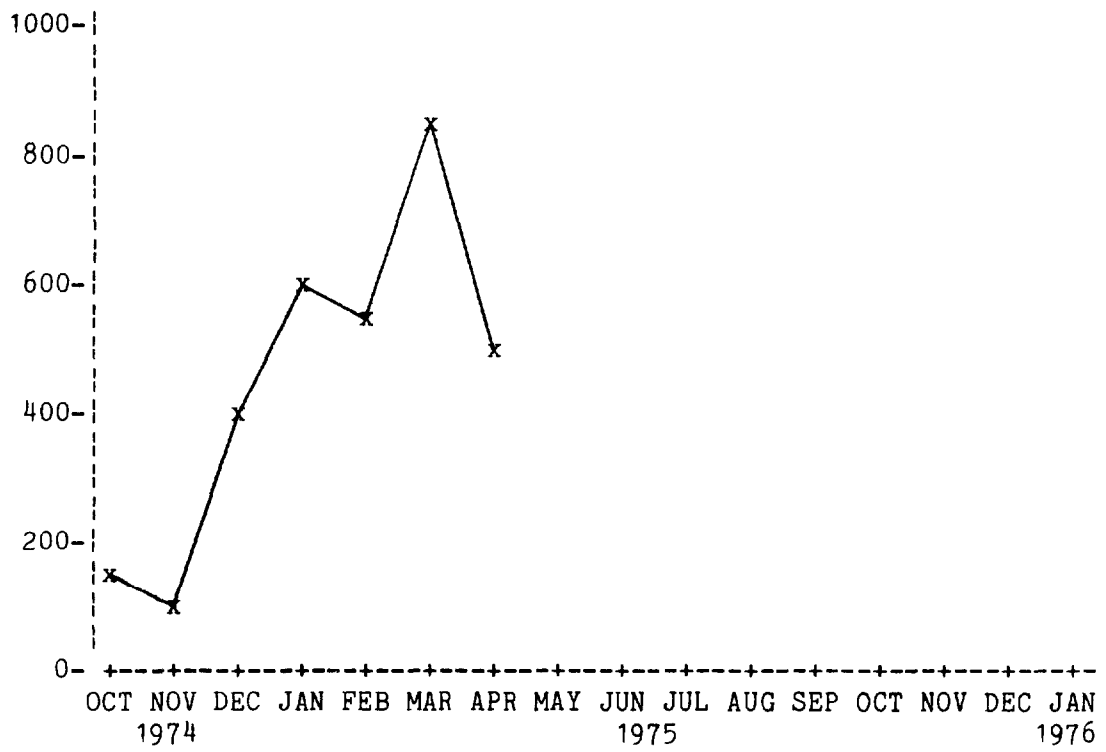
### II.B.3 NETWORK USAGE STATISTICS

#### NETWORK USAGE PLOTS

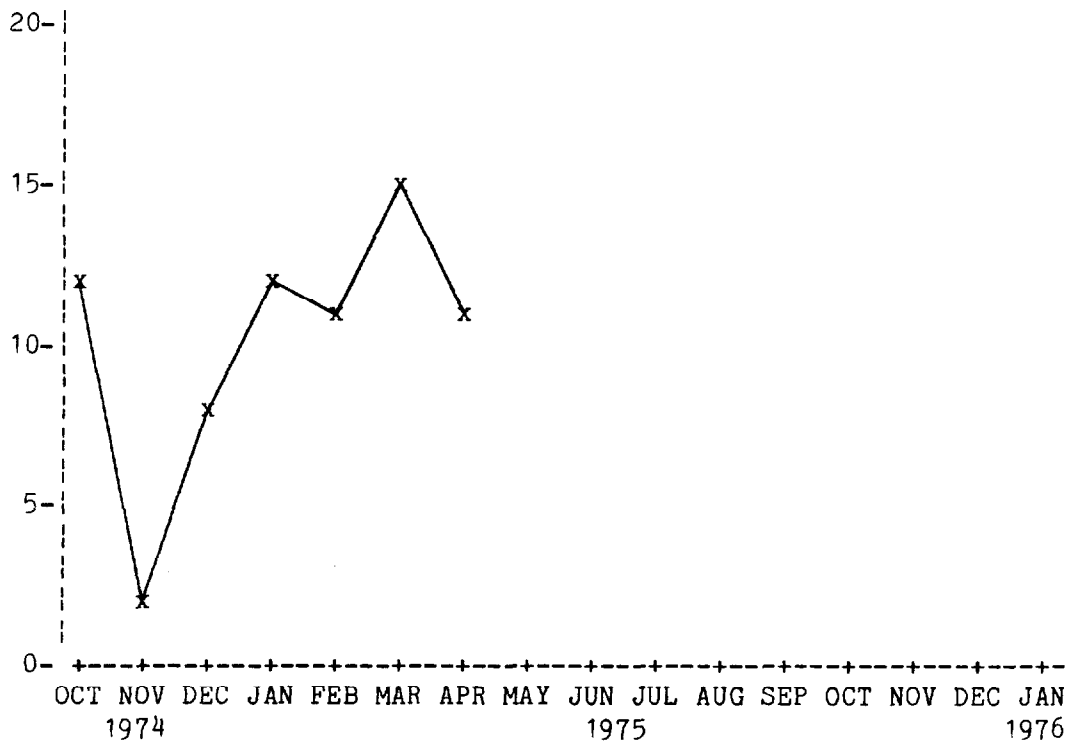
The following are plots of the major billing components for SUMEX-AIM TYMNET usage. These include the total connect time for terminals coming into SUMEX and the total number of characters transmitted over the net. The ratio of characters received at SUMEX to characters sent to the terminal is about 1:14 over the past couple of months. The plot on page 3 is for TOTAL character traffic.

Note that the high usage in October is because we were doing a great deal of testing of the net at that time. As of March, we announced that the IN-WATS lines would be terminated soon and that users should transfer to the TYMNET. Extrapolation of these data should be done very carefully because the apparent steep rise in February and March has apparently been tempered by the ARPANET connection becoming operational in late March. The Rutgers community, at least in part, is using the ARPANET in place of the TYMNET because of more convenient access through their TIP.

TYMNET CONNECT TIME (IN HOURS)



TYMNET CHARACTER TRANSMISSION (MILLIONS OF CHARS)



#### II.B.4 SYSTEM DIURNAL LOADING PROFILE

Since late February, we have been collecting more detailed information about time variations in system performance and loading. These data include user-oriented measurements such as load average, number of jobs, and percent of LISP use. In addition, we collect data characterizing internal system operating parameters such as paging rate, trap overhead, scheduler overhead, and drum use. Systematic measurements are taken every 20 minutes to give an overall picture of diurnal variations. Periodically we observe system dynamics on a much shorter time scale as well to observe more of the internal workings of the system - such data are so voluminous, however, that they cannot be collected continuously.

The following graphs give a feeling both for the type of data available and the loading characteristics they reveal. The plots are organized 2 per page with the quantity displayed labeled at the top of each graph along with average and extremum values of the data. The first 3 pages display composite data for the month of April 1975 - derived by averaging together all of the weekday data for that month as a function of time of day. The next 3 pages show somewhat different data for a single day recently in May (note the hash marks on the right side of these plots signify that the machine was down [for repair]).

The most striking feature from the monthly data is the expected peak loading during prime time and in fact a bi-modal distribution is apparent in some of the plots approximating the 3 hour shift between prime times on the east coast and west coast. Another striking feature is the dramatic difference between the monthly average data for April and the individual daily data. We feel there are at least two effects taking place to account for this difference in addition to simple day-to-day loading level variations. First, daily data is considerably more "noisy" than the average data with peak loading occurring in cycles spaced about 2-3 hours apart. We observe the phases of the various sub-peaks relative to time of day to be unpredictable other than gross effects leading to the bi-modal average data corresponding to heavy mid-afternoon use of the respective coasts. For example, whereas the peak in the monthly load average curve occurs around 1500 PDT, the peak for the daily plot is around 1600-1700 and the daily plot displays much higher "momentary" (20 minute average) excursions. In fact, over shorter intervals (1-2 minutes) load average peaks are even higher (load averages approaching 10 or more are not uncommon under present loading in the afternoon). Thus in taking the average for a month, the peak loading extremes which strongly influence the subjective "feel" of the system on a daily basis are blurred out and lost.

Secondly, we are apparently on a rather steeply rising curve in terms of system utilization. The composite load average data for March, for example, had a peak value of 2.7 as opposed to 3.7 for April. Of course, two points do not necessarily constitute a trend but this data corresponds with the subjective comments contributed by users in their project reports and other commentaries on system response. There is general agreement that loading has been increasing noticeably.

Several points mentioned in the report can be observed in the daily loading data. During peak loading, drum use somewhat exceeds the 3300 page fixed head capacity we now have on-line. We expect to mitigate this overflow through more intelligent management of swapping storage. The total number of jobs follows closely the average data plots but other measures of system use like load average and LISP job consumption tends to skew toward the evening hours. This is evidently a natural leveling process wherein the very large jobs have more difficulty contending during the day and the program developers have shifted their schedules to work later at night. Observe the very heavy (and efficient - 96%) use during the evening hours when the load average has dropped to about 3 or 4 and is arising to a large extent from LISP usage. It should also be noted that load average and overall percent usage are somewhat independent. Usage measures how many cycles are being consumed and load average measures between how many users these cycles are being divided (one CPU-bound job would show a usage of 100% but a load average of only 1.0).

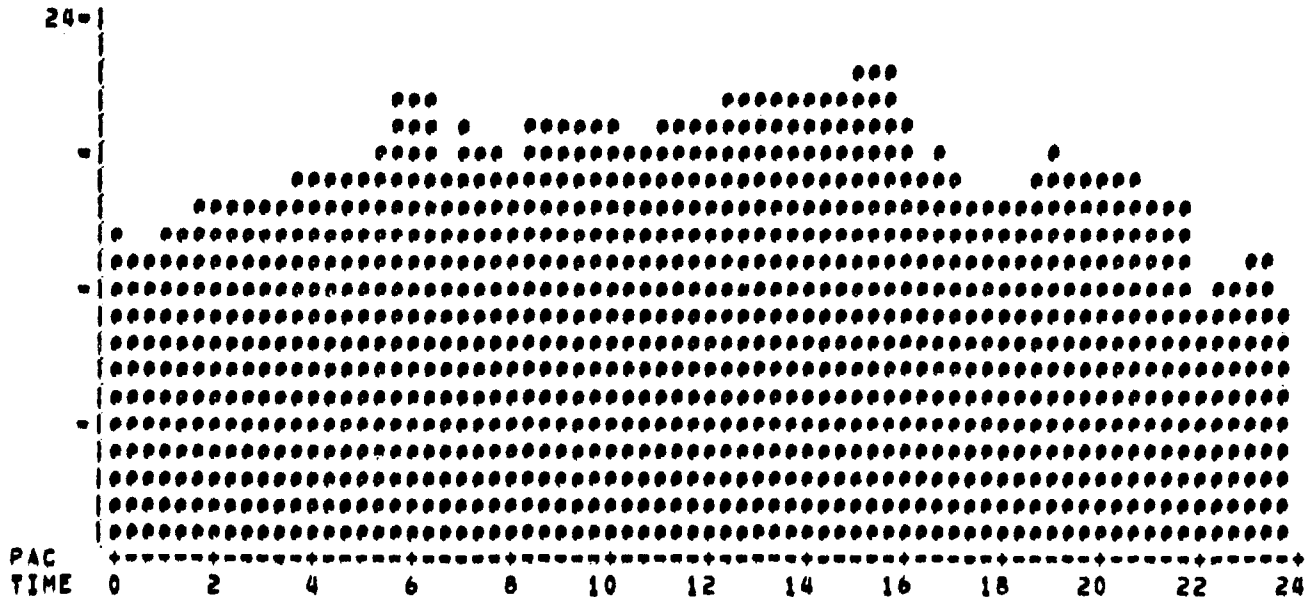
These data, while preliminary in the sense that we have only the trends of a couple of months to go on, are consistent with user comment and are the basis of our preliminary proposal to up-grade the SUMEX-AIM CPU (see page 4 AND Appendix C). While load-leveling (such as moving work on LISP programs to evening hours) may be acceptable (and desirable) for some program development work, it is unrealistic to ask physicians to adapt in a similar manner in trying out the AI programs. Our development and managerial incentives are directed toward making better use of the off-hours, freeing up prime time where possible. But, we must anticipate the need for more computing capacity during prime time, especially based on these measurements and the anticipated growth of the user community. We are continuing active work on the CPU up-grade plan discussed elsewhere in this report (see page 51).

## AVERAGE DATA PLOTS - WEEKDAYS, APRIL 1975

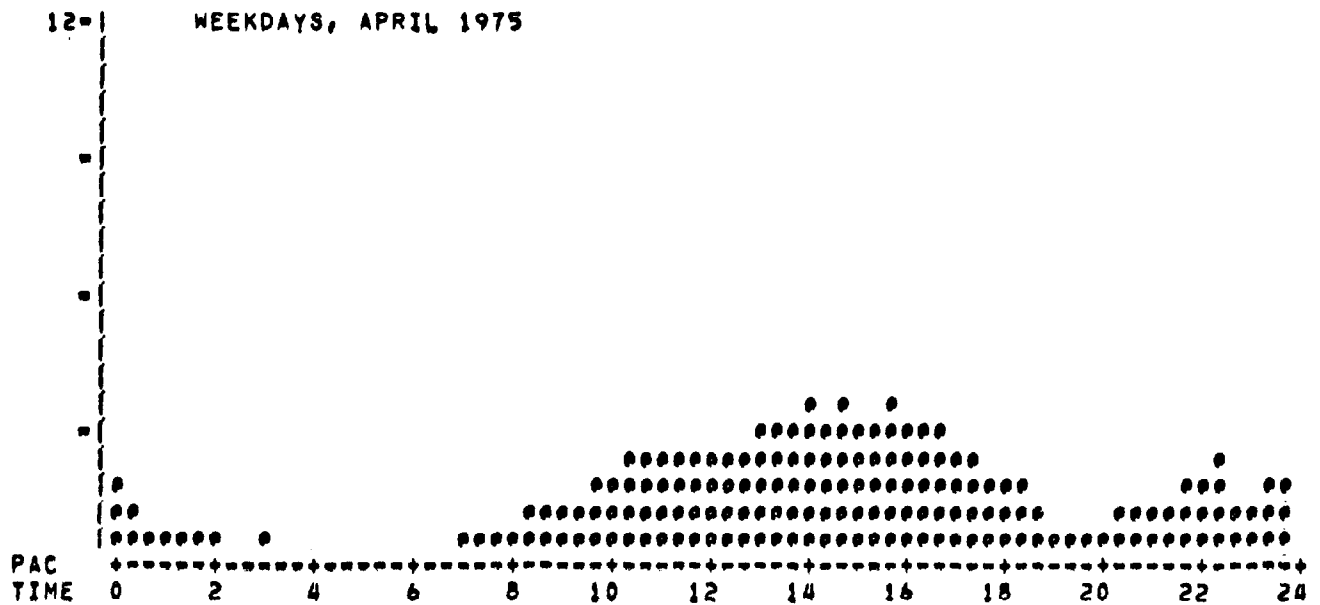
Includes dates:

TUESDAY, 1-APR-75  
 WEDNESDAY, 2-APR-75  
 THURSDAY, 3-APR-75  
 FRIDAY, 4-APR-75  
 MONDAY, 7-APR-75  
 TUESDAY, 8-APR-75  
 WEDNESDAY, 9-APR-75  
 THURSDAY, 10-APR-75  
 FRIDAY, 11-APR-75  
 MONDAY, 14-APR-75  
 TUESDAY, 15-APR-75  
 WEDNESDAY, 16-APR-75  
 THURSDAY, 17-APR-75  
 FRIDAY, 18-APR-75  
 MONDAY, 21-APR-75  
 TUESDAY, 22-APR-75  
 WEDNESDAY, 23-APR-75  
 THURSDAY, 24-APR-75  
 FRIDAY, 25-APR-75  
 MONDAY, 28-APR-75  
 TUESDAY, 29-APR-75  
 WEDNESDAY, 30-APR-75

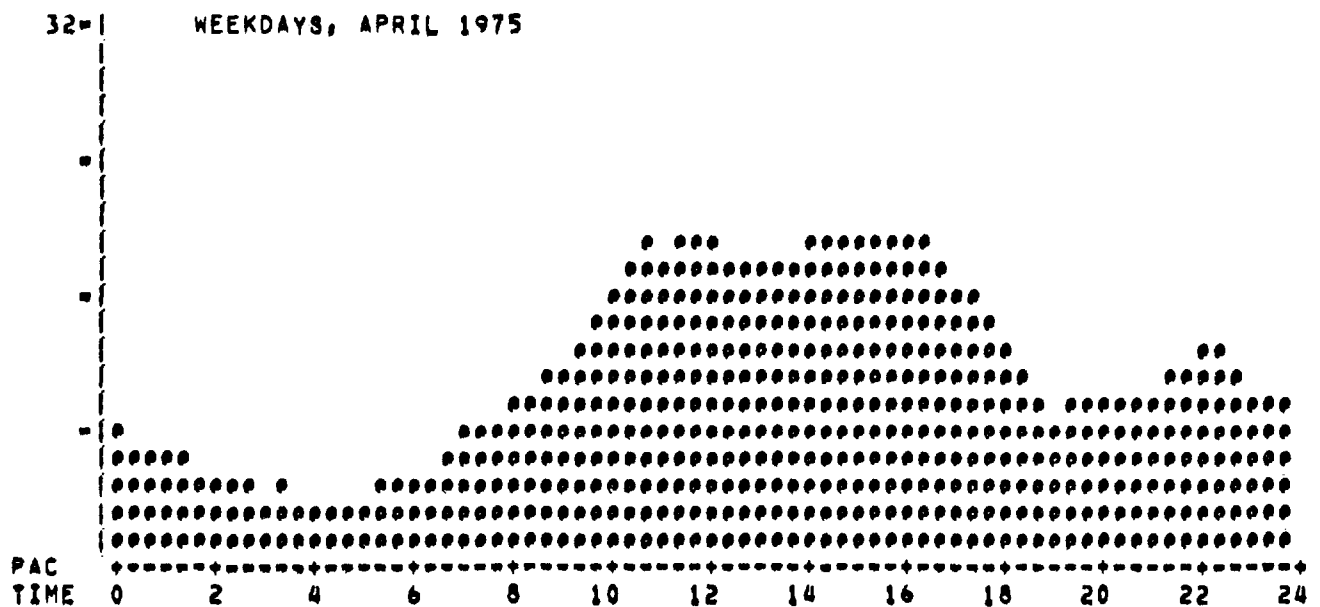
CUMULATIVE UPTIME: Number of Days Data in Average Plots



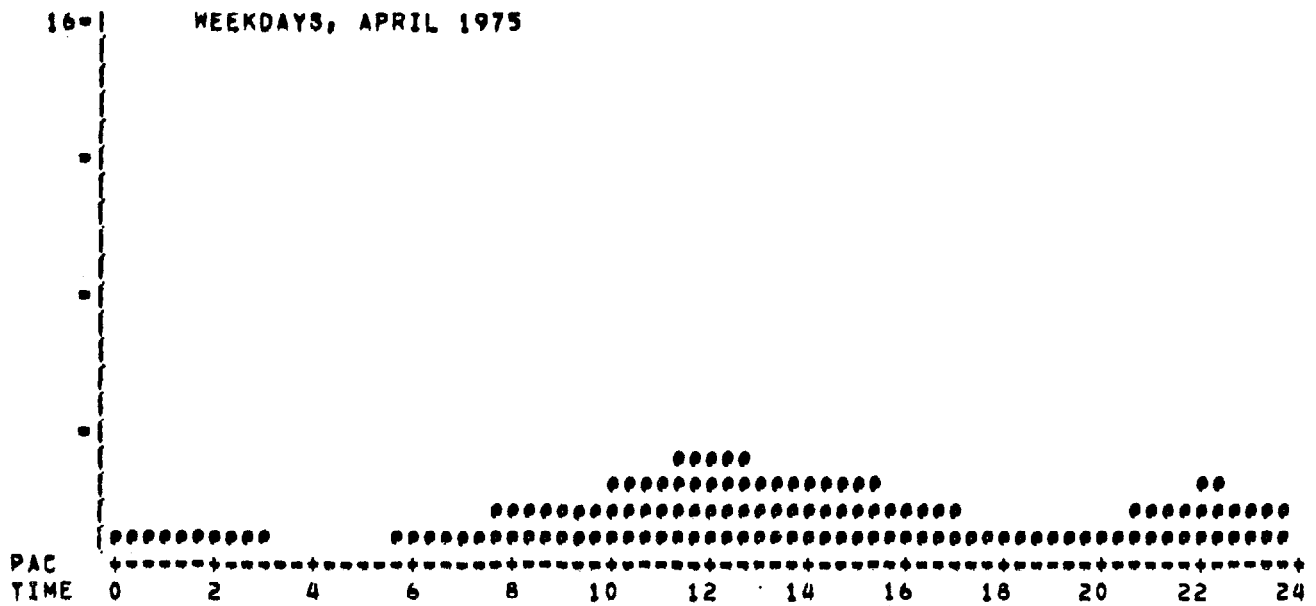
AVERAGE DATA: Load Average (Low= .0, Ave= 1.5, High= 3.7)



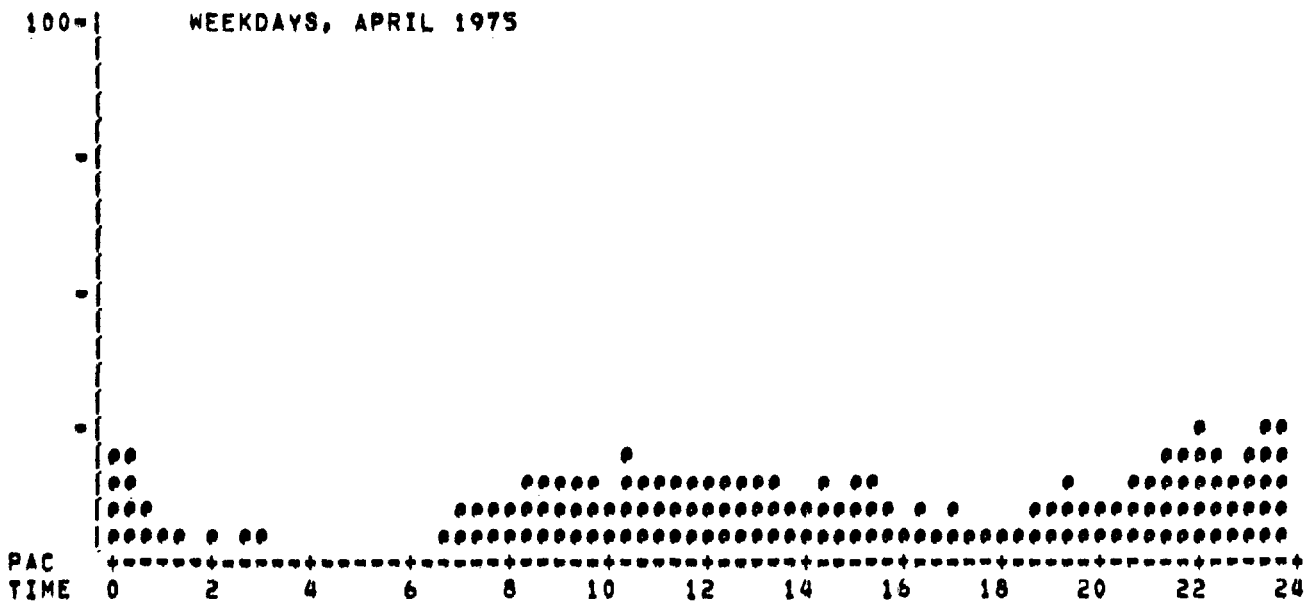
AVERAGE DATA: Total Number of Jobs (Low= .0, Ave= 11.2, High= 19.8)



AVERAGE DATA: Number of LISP Jobs (Low= ,0, Ave= 1.4, High= 3.2)

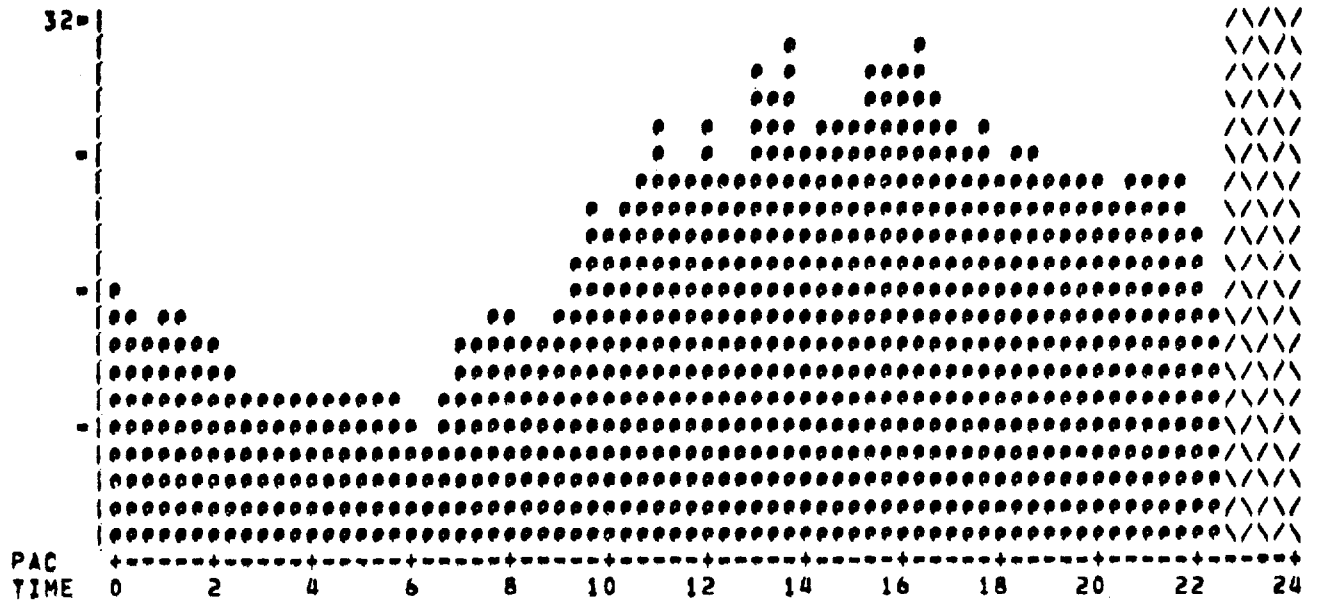


AVERAGE DATA: % LISP Usage (Low= ,0, Ave= 10.5, High= 24.0)

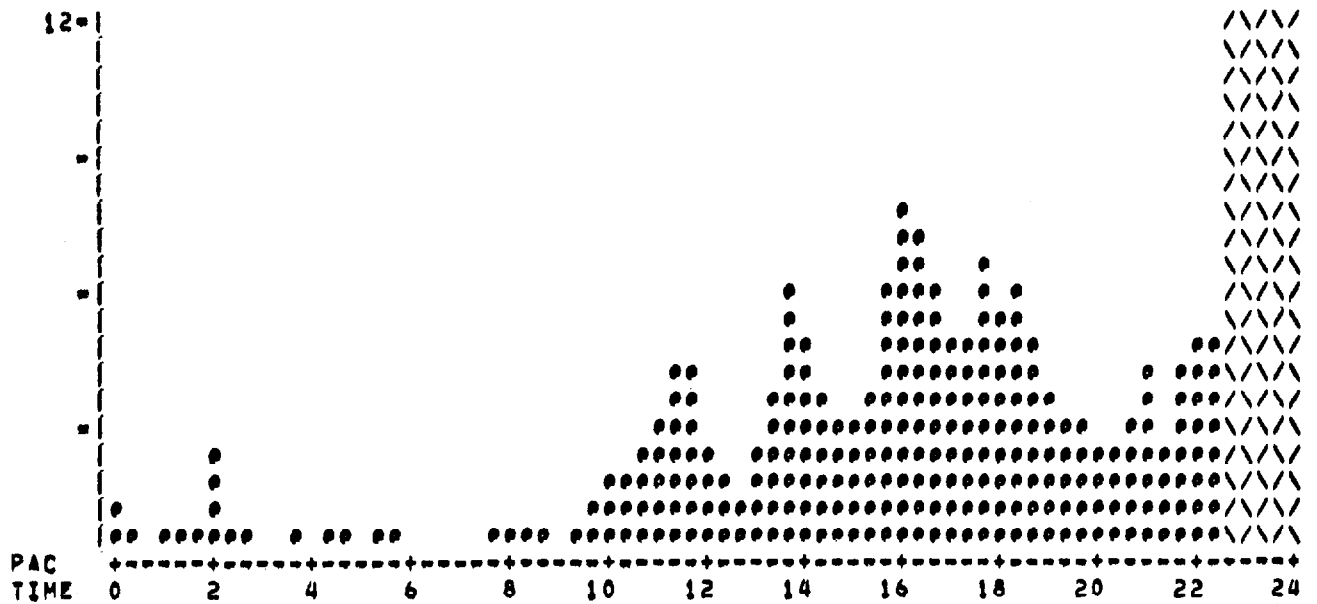




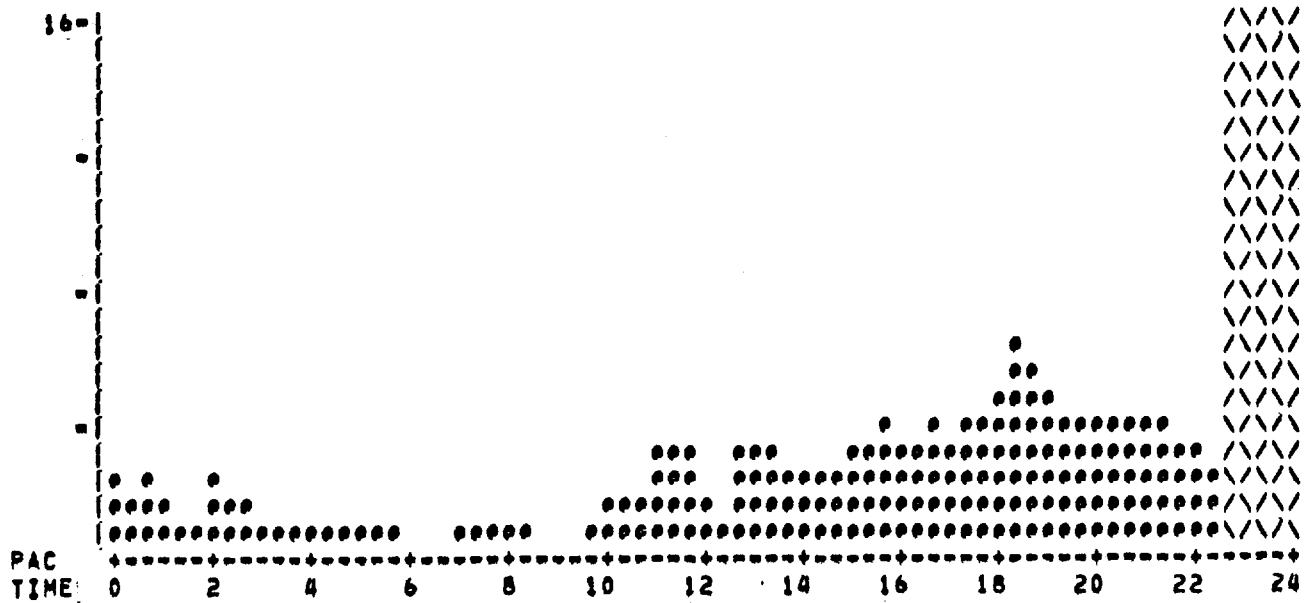
MONDAY, 19-MAY-75: Total Number of Jobs (Low= 6,7, Ave= 18,5, High= 30,7)



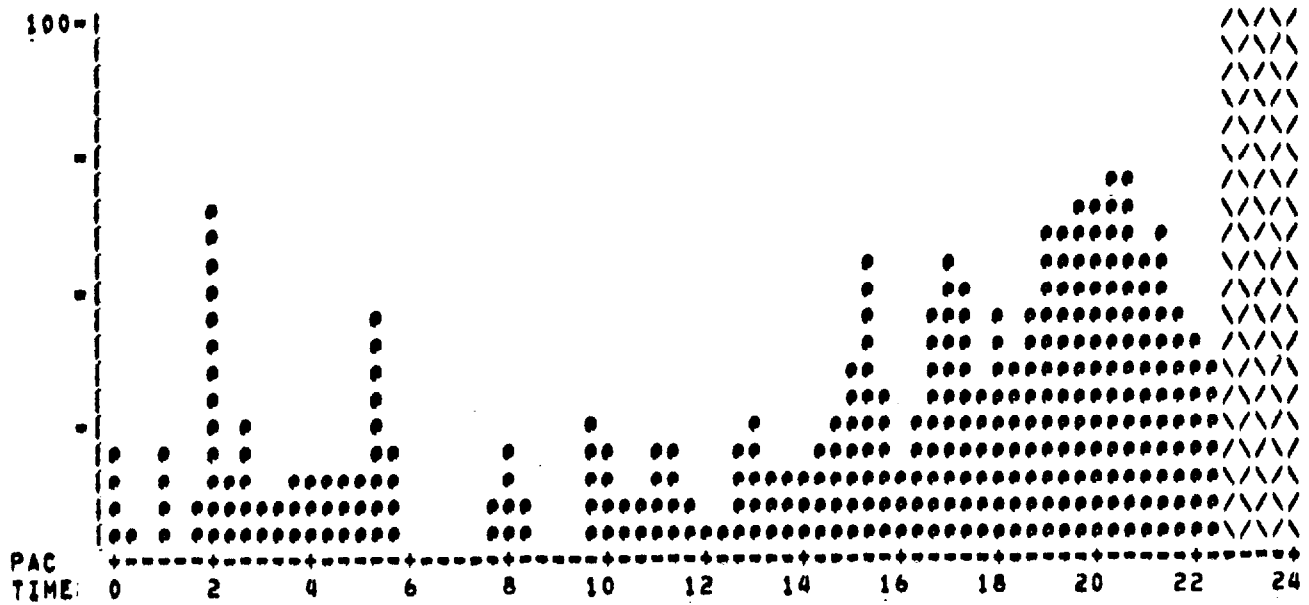
MONDAY, 19-MAY-75: Load Average (Low= ,0, Ave= 2,4, High= 7,7)



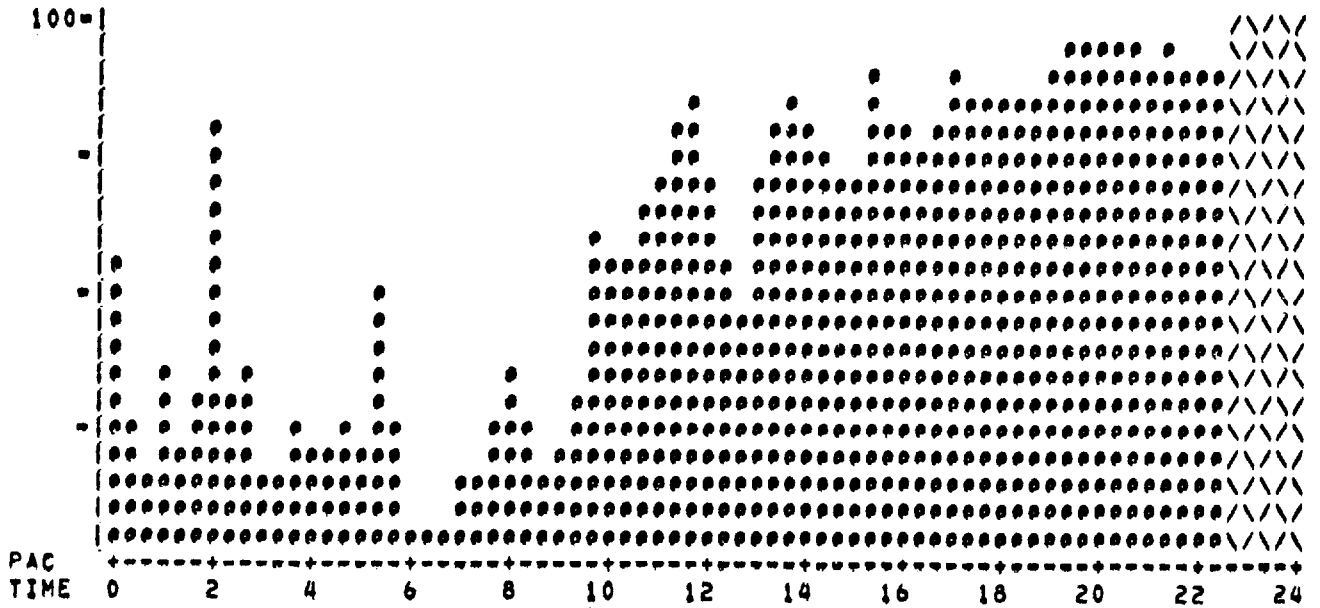
MONDAY, 19-MAY-75: Number of LISP Jobs (Low= ,0, Ave= 2,1, High= 5,9)



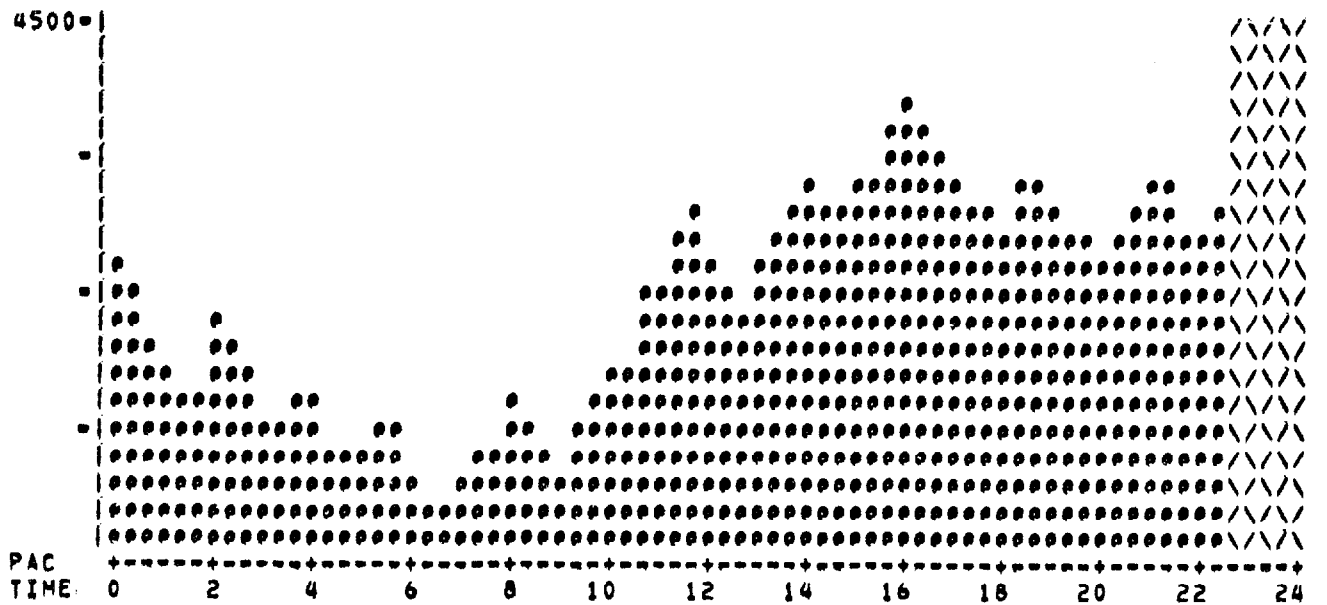
MONDAY, 19-MAY-75: % LISP Usage (Low= ,0, Ave= 25,0, High= 72,0)



MONDAY, 19-MAY-75: % Time Used (Low= 2,0, Ave= 56,0, High= 96,0)



MONDAY, 19-MAY-75: Drum Pages in Use (Low= 405,0, Ave= 2099,1, High= 3690,0)



## II.C RESOURCE EQUIPMENT SUMMARY

The following table gives a list of the items of equipment purchased to date for the SUMEX resource along with details on vendor, description, price, and date.

ITEM	QUANTITY	DESCRIPTION	MANUFACTURER	MODEL NUMBER	DATE INSTALLED	DATE ACCEPTED	PURCHASE PRICE	SOURCE FUNDS
KI-10 CPU	1	Central processor, including console	Digital Equipment Corporation	KI-10	3/1/74	4/24/74	\$178,500	NIH
Memory	3	Core memory (64K words including 4 MC-10 memory parts	Digital Equipment Corporation	MF-10G	3/1/74	4/24/74	\$224,910	NIH
	1	Core memory (64K words including 4 MC-10 memory ports	Digital Equipment Corporation	MF-10G	11/74	12/74	\$ 63,754	NIH
	1	Memory port multiplexer	Digital Equipment Corporation	MX-10	8/74	9/74	\$ 4,770	NIH
Clock	1	Programmable clock	Digital Equipment Corporation	DK-10	3/1/74	4/24/74	\$ 2,678	NIH
Disk System	1	Single double density disk controller	Digital Equipment Corporation	RP-10C	3/1/74	5/1/74	↓	↓
	1	Memory data channel	Digital Equipment Corporation	DF-10	3/1/74	4/24/74		
	4	Double density disk drives and disk packs	Digital Equipment Corporation	RP-03	3/1/74	4/24/74	\$108,153	NIH
	3	Double density disk drives and disk packs	Digital Equipment Corporation	RP-03R	2/75	3/75	\$ 44,636	NIH

ITEM	QUANTITY	DESCRIPTION	MANUFACTURER	MODEL NUMBER	DATE INSTALLED	DATE ACCEPTED	PURCHASE PRICE	SOURCE FUNDS
Swapping Storage	2	Fixed head disk with 1.7M word capacity and 4 track parallel access	Digital Development Corporation	A-7312-D-8	1/75	3/75	\$ 37,206	NIH
	1	Special systems controller for DDC disks	Digital Equipment Corporation	RES-10	10/74	11/74	\$ 81,090	NIH
DEC Tapes (TU-56)	1	DEC tape control	Digital Equipment Corporation	TD-10	3/1/74	4/24/74	↓	↓
	1	Dual DEC tape drive	Digital Equipment Corporation	TU-56	3/1/74	4/24/74	\$ 17,850	NIH
Magnetic Tapes (2 x TU-30)	1	Magnetic tape controller	Digital Equipment Corporation	TM-10A	3/1/74	4/24/74	↓	↓
	2	Tape transports	Digital Equipment Corporation	TU-30	3/1/74	4/24/74	\$ 31,238	NIH
Line Printer	1	Special systems line printer control for Data Products 2410	Digital Equipment Corporation	Special	6/74	7/74	\$ 7,208	NIH
	1	Line printer with 96 character drum, vertical format control, parity check	Data Products	2410	6/74	7/74	\$ 18,963	NIH

ITEM	QUANTITY	DESCRIPTION	MANUFACTURER	MODEL NUMBER	DATE INSTALLED	DATE ACCEPTED	PURCHASE PRICE	SOURCE FUNDS
GT-40	1	Graphics terminal	Digital Equipment Corporation	GT-40	3/1/74	4/24/74	\$ 11,156	NIH
Line Scanner	1	Data line scanner	Digital Equipment Corporation	DC-10A	3/1/74	4/24/74	↓	↓
	1	8-line unit	Digital Equipment Corporation	DC-10B	3/1/74	4/24/74	\$ 16,275	NIH
TYMNET Interface	1	PDP-10 TYMNET communications controller	TYMSHARE		8/74	10/74	\$ 50,774	NIH
ARPANET Interface	1	BB&N ARPANET/KI-10 interface	Bolt, Beranek & Newman		1/75	2/75	\$ 21,200	NIH
PDP-11/10	1	Communications processor	Digital Equipment Corporation	PDP-11/10	2/75	3/75	\$ 13,445	NIH

ITEM	QUANTITY	DESCRIPTION	MANUFACTURER	MODEL NUMBER	DATE INSTALLED	DATE ACCEPTED	PURCHASE PRICE	SOURCE FUNDS
Terminals	1	Terminal	Data Terminals Communications	DTC-300	3/18/74	6/74	\$ 4,597	NIH
	2	Terminals - Execuport portable with carry case	Computer Transceiver Systems, Inc.	311-3	3/18/74	6/74	\$ 6,402	NIH
	6	Terminals - elite CRT with edit capabilities	Datamedia	2500	9-10/74	11/74	\$ 13,250	NIH
Keyboards	3	Keyboards, special, for leased Datamedia elite 2500 CRT terminals at - NIH Johns Hopkins Univ. Washington Univ.	Datamedia	70DVK7019	11/74	12/74	\$ 1,156	NIH



ITEM	QUANTITY	DESCRIPTION	MANUFACTURER	MODEL NUMBER	DATE INSTALLED	DATE ACCEPTED	PURCHASE PRICE	SOURCE FUNDS
Modems	16	Auto answer modems	Prentice Electronics	P-113B	5/6/74	5/6/74	↓	↓
	5	Auto answer modems	Prentice Electronics	P-1200/150	5/6/74	5/6/74		
	5	Originate modems	Prentice Electronics	P-1200/150	5/6/74	5/6/74		
	4	Modem enclosure with loopback switch and cables	Prentice Electronics	P-100	5/6/74	5/6/74		
	4	Modem enclosures for 8 modems with cables, power supply, digital loopback, line loopback, indicator lights	Prentice Electronics	P-850	5/6/74	5/6/74		
	2	Acoustic coupler modems	Prentice Electronics	DC-22	3/74	3/74	↓	↓
	3	Modem enclosure with live loopback switch to house P-103F modems	Prentice Electronics	P-100	3/74	3/74	\$ 9,415	NIH
Oscilloscope	1	Oscilloscope	Tektronix, Inc.	475DM43	1/75		\$ 3,476	NIH

## II.D PUBLICATIONS

Publications for the SUMEX staff have included papers describing the SUMEX-AIM resource coincident with its dedication last November (SIGART Newsletter, Sept. 1974; and ACM SIGBIO meeting, San Diego, November 1974 (oral presentation)). In addition, a substantial effort has gone into developing, upgrading, and extending documentation about the SUMEX-AIM resource, the SUMEX-TENEX system, and the many subsystems available to users. These efforts include a number of major documents (such as SOS, PUB, and TENEX-SAIL manuals) as well as a much larger number of document upgrades, user information and introductory notes, and policy guidelines (see Appendix E, Appendix H, and Appendix I). Publications for individual user projects are summarized in the respective reports and Appendix F, page 167.

### III RESOURCE FINANCES

#### III.A REFERENCE TO BUDGETARY DETAILS

The budgetary materials for the SUMEX project covering past actual costs, current performance, and estimates for the next grant year are submitted in separate document to the NIH. Only that section referenced earlier and describing preliminary plans for SUMEX-AIM CPU up-grading are included here

### III.B PRELIMINARY SUMEX-AIM CPU AUGMENTATION PLAN

#### PRELIMINARY PLAN FOR AUGMENTING SUMEX-AIM TO KL-10

We have indicated a budgetary provision of \$132,051 as funds carried forward from year 02 for system augmentation. As discussed in the report (See "SYSTEM DEVELOPMENT - PDP-10 HARDWARE", page 4), we have been observing system performance over the past few months as the user load has increased and find that the response time loading during prime shift approaches saturation. Based on these data and the discussions in Appendix C, we feel that the next most significant bottleneck to system throughput for the AIM community will be in the central processor. We are very concerned about being able to accommodate adequately both the continuing development of the AI performance programs and the exposure of working physicians in a meaningful way. Of course, the NON-prime time loading does not yet approach saturation: we are working on both technical and managerial incentives to encourage use of off-hours (a surprising number of INTERLISP users already are working in the evening and night to get a less loaded machine). However, the very nature of interactive computing in consultative programs is that human beings are involved and because of other commitments for professional people (e.g., physicians), their main load in using the machine will be during prime time. We must then take some action to increase system throughput to assure satisfactory user response time under the increasing prime time load and our judgement is that the most effective augmentation would be in CPU capacity.

Over the past year a potential solution to this problem has emerged with DEC's announcement of the KL-10 processor, scheduled to begin delivery this summer of 1975. The KL-10 is a microprogrammed machine using a high speed cache memory and inherently faster logic to achieve a throughput estimated by DEC to be 2-3 times that of the KI-10. The relationship of DEC's estimates to the performance of a KL-TENEX with many INTERLISP jobs is not easily evaluated, particularly in view of needed additional detailed information described below. However, for a relatively small cost increment (15%) over the capital investment in the facility, an increased throughput of around 2 times can be expected. In a recent example of where the IMSSS system was up-graded from a KA-10 to a KI-10 (1.8 times as fast), the load average for a comparable job mix dropped from 15-20 down to 5-10. In view of a number of uncertainties about this solution at this time which must be resolved, we propose that unobligated year 02 funds be carried forward and held in reserve by BRB until the necessary technical and administrative decisions can be made on a more detailed proposal to be submitted some months hence. At that time, we also anticipate an enlargement of our user base which would simultaneously increase a) the current loading and b) the justification for remedial investment.

From a technical viewpoint, KL-10's have not been delivered yet and available documentation is quite superficial. We do not want serial number 1 as we are heavily committed to providing RELIABLE

community computing resources. The lack of precise documentation makes it difficult to more than cursorily assess the problems in transferring TENEX to the KL-10. At this time it appears that the KL will look initially (i.e., with the initial microcode) like a KI-10 in terms of page faulting and other functions. This would make mounting TENEX quite straightforward. Other questions arise, however, about the management of system loading and diagnostics. DEC's approach with regard to TOPS-10 is to interface the system utility PDP-11/40 (integral to the KL-10) with the new RP-04 disk hardware for this purpose. One of these drives with controller would cost over \$80,000 alone. DEC is still considering our request for information about supporting other disk systems (as the PDP-11 can easily do) and supplying necessary information about the PDP-11 programming to allow us to modify it. Without such necessary technical information we are delayed in doing the detailed planning. Based upon current list prices from DEC, the cost of upgrading to a KL-10 would be about \$150,000. This price is somewhat above the the amount which apparently can be carried forward (additional funds may be available if they do not have to be spent on the remaining ARPANET interface work) but in light of our previous success in working with DEC and achieving a discount, this price may be brought into range of available funds.

## III.C RESOURCE FUNDING

The SUMEX-AIM resource is essentially wholly funded by the Biotechnology Resources Branch [\*]. The various collaborator projects which use SUMEX are independently funded with respect to their manpower and operating expenses. They obtain from SUMEX, without charge, access to the computing and, in most cases, communications facilities in exchange for their participation in the scientific and community building goals of SUMEX.

[\*] Except for the participation by Stanford University in accordance with general cost-sharing, and for assistance to SUMEX by other projects with overlapping aims and interests.

#### IV RESOURCE PROJECT DESCRIPTIONS

The following are inputs from the various user projects currently in the SUMEX-AIM community. These project descriptions and comments are the result of a solicitation for contributions sent to each of the project Principal Investigators requesting the following information:

"Please submit for the SUMEX-AIM annual report:

- 1) Updated abstract of project goals and activities. We plan to assemble these on-line as well for community information.
- 2) A summary of project accomplishments over the past year achieved by means of SUMEX. Please include references to any publications which have resulted.
- 3) Comments and assessments of your experiences in interacting with the SUMEX resource. These should include any successes or problems and include technical aspects as well as community or administrative aspects of your collaboration.

It is also important that you indicate to us how your use or non-use of the system corresponded to your expectations and agreements with us at the time you were enrolled as a SUMEX user. This is the time to report extensions of the scope of your project, as well as pilot trials that you may have initiated. We encourage such experimentation without great formality, but we expect that departures that involve significant computer usage will be reported to us at this time for re-review."

The text which follows on the various projects is primarily the responsibility of the indicated project leaders.

#### IV.A     FORMALLY APPROVED PROJECTS

##### IV.A.1     STANFORD USERS

##### IV.A.1.a     DENDRAL PROJECT

Principal Investigators: Profs. C. Djerassi (Chemistry),  
J. Lederberg (Genetics), and E. Feigenbaum (Comp. Sci.)

(Grant NIH RR-00612-05, 3 years, \$323,933 this year)

#### OVERVIEW

The Heuristic DENDRAL project is an application of artificial intelligence to biomedical molecular structure determination problems. Under NIH funding the project has moved significantly closer to making the computer programs and structure elucidation techniques available to a broad community of scientists. This brief report is organized in three parts according to the three major aims of the project: (PART 1) Enhancing the power of the mass spectrometry resource, (PART 2) Developing performance and theory formation programs, and (PART 3) Applying the computer programs and instrumentation to biomedically relevant structure elucidation problems.

The highlight of the period since May 1, 1974, was the project's move to the interactive computing environment of the NIH-funded SUMEX-AIM facility from the batch computing environment of the Stanford Computation Center. Because of this, many scientists outside this university have been able to use the DENDRAL computer programs for their own research. Also, the programs themselves grew in power and scope, and we opened new vistas for collaboration with other research groups. We have been able to make the programs more conversational and thus more helpful to the chemists and biochemists for whom they were developed. Outside users in other research groups also have in SUMEX an easy mechanism for trying out the DENDRAL programs on their own structure elucidation problems. Finally, we have a mechanism for looking at subroutines developed by other research groups in the context of our own programs -- and have incorporated subroutines written, for example, by T. Wipke and by R. Feldmann, into our procedures. The programs and their development are discussed in Part 2, below.

The DENDRAL project, one of the major users of the SUMEX-AIM computer facility, has been forming its own community of remote users. This "exodendral" community has already provided valuable contributions to program development and both the community and contributions are expected to grow at an increased rate. As an example, for the last month for which figures are available (March 1975), the number of CPU hours used by exodendral persons amounted to at least ten percent of the CPU hours used by the entire DENDRAL project which, of course, still reflects heavy CPU-intensive



development efforts. In the last month alone, one new exodendral account representing at least three users has been added to the system, and another four exodendral users have been invited to begin their usage via various "guest" accounts.

Our programs are receiving heavy use from local users and outside users who are investigating mass spectrometry problems for a variety of different compound classes. In addition, new program developments have extended the scope of biomedical structure elucidation problems for which we can provide some computer assistance. Local users include members of Professor Djerassi's group, other chemistry department persons and research groups at the Stanford Medical School. We have recently begun the process of building a community of outside users who can access our programs at SUMEX via TYMNET or ARPANET. Several research groups have expressed considerable interest; we have demonstrated and explained the programs to several groups and we are currently arranging more demonstrations and assisting other people in learning to use SUMEX and the programs from their own laboratories. These applications are discussed in detail in Part 3, below.

#### PART 1: ENHANCING THE POWER OF THE M.S. RESOURCE

Our grant proposal requested funds for significant upgrading of our capabilities in mass spectrometry. The goals of this upgrading were to provide routine high resolution mass spectrometry (HRMS), combined gas chromatography/low resolution mass spectrometry (GC/LRMS) and to develop a combined gas chromatography/high resolution mass spectrometry (GC/HRMS) facility. In addition, this would provide the capability for new experiments in the detection and utilization of data on metastable ions. These capabilities would then be available as required for application to our wider goal, solution of biomedical structure elucidation problems of community of researchers.

The upgrading included several items of hardware and software development, as follows: 1) Acquire stand-alone computer support for the mass spectrometer because existing facilities were inadequate and very expensive; 2) convert existing software, written in the PL/ACME language into FORTRAN so that it would run on the new system; 3) develop new software as required for the demanding task of GC/HRMS; 4) provide hardware and software for semi-automatic acquisition of data on metastable ions. The initial development phase of this upgrading included performance tests to determine the capabilities and limitations of the GC/HRMS system to define the scope of problems to which it can be applied.

#### PART 2: DEVELOPING PERFORMANCE AND THEORY FORMATION PROGRAMS TO ASSIST IN BIOMEDICAL STRUCTURE ELUCIDATION PROBLEMS

The Heuristic DENDRAL computer programs assist with structure elucidation problems by helping interpret mass spectra and helping generate structures that are consistent with the interpretations. The Meta-DENDRAL programs assist with rule formation problems in cases where the rules of mass spectrometry are not known.

All programs have been transferred to the SUMEX machine and most have been considerably improved from their previous versions. The CONGEN and PLANNER programs, in particular, have been improved substantially because these two were thought to offer the most to scientists with structure elucidation problems. Two new programs were developed in this period: CLEANUP and MOLION. The CLEANUP program helps separate the mass spectra of individual components from a GC/MS analysis, and eliminates the background due to GC column "bleed". The MOLION program determines the mass and empirical formula of the whole molecule from its mass spectrum, without prior knowledge of any of the features of the molecule. Both of these new programs solve major problems that had to be determined manually previously before a scientist used the DENDRAL programs.

### PART 3: APPLICATIONS TO BIOMEDICAL STRUCTURE ELUCIDATION PROBLEMS

One of our major aims is to apply the instrumentation and computer programs described above to the study of molecular structure problems in a variety of biomedical applications areas. In order to do that we have made it quite clear that our facilities would be made available to wider community of collaborators/users as our resources permitted. Both categories of application, i.e., within our own group, and with an outside group, are described below.

We have taken several steps toward encouraging a broad community of potential users to call on our facilities. For example, we sent a memorandum to local persons who had indicated their potential need for our facilities as described in our proposal. A questionnaire sent to members of the American Society for Mass Spectrometry, Committee III on Computer Applications, resulted in about 55 persons indicating a desire to know more about access to our programs. A subsequent description of the DENDRAL programs was sent to these persons and to several other persons whom we know from personal contacts might be interested. Because of the nature of their investigations, many of these people receive NIH support.

The availability of SUMEX as a mechanism for resource sharing has made it possible for us to extend access to our programs to a number of people. Without SUMEX, this access would be impossible, and most of our programs (especially LISP routines which are not easily exportable) could be used only by ourselves.

#### Applications by Professor Djerassi's Research Group

Prof. Djerassi's research group at Stanford makes extensive use of the computer programs and the mass spectrometry facility. These applications are detailed in the DENDRAL annual report to the NIH.

#### Applications by Other Members of the Stanford Chemistry Dept.

- 1) Prof. Mosher: We have used CONGEN to suggest structural possibilities for a naturally occurring analog of the fish poison tetrodotoxin. This structure is still under investigation.
- 2) Prof. Hahn (on sabbatical leave at Stanford from Syracuse University): We have used CONGEN to explore possible structures for unknown products of a photochemical reaction. These results have led him to begin a new set of experiments (specifically, CMR) to greatly restrict the possibilities.
- 3) Prof. Johnson: In his wide-ranging syntheses of steroid hormones and other steroids of biological interest, he has studied reactions involving stereo-specific cyclization. We are investigating use of CONGEN for structural analysis under constraints imposed by synthetic cyclization experiments. For example, a previously investigated compound was found to have two structural possibilities. The new possibility could not be differentiated from the assigned structure based on available data.
- 4) Prof. Collman: We have utilized our mass spectrometry facilities to analyze samples in support of his work on oxygen binding to porphyrins (hemoglobin models).
- 5) Prof. Van Tamelen: We have provided mass spectrometry support (HRMS) to assist in the characterization of several compounds related to his work on terpenoid cyclizations.

#### Applications by Other Stanford University Scientists

- 1) Genetics Research Center (GRC) Stanford Hospital: One of our strongest collaborations because of their requirements for additional automation in data reduction and analysis. Their screening program for metabolites characteristic of diseases of genetic origin uses GC/LRMS as the primary source of data. The CLEANUP and MOLION programs were written at least in part to assist the GRC in more systematic approaches to their data. We are currently using CONGEN to assist in determination of structures of unknowns for which mass spectrometric and chemical data are available. Our GC/HRMS facilities will also be utilized for problems which require determination of empirical formulas for ions in spectra of unknown compounds.
- 2) Stanford Pharmacy: We have had several requests for assistance from the Pharmacy of Stanford Hospital (Director: Dr. Hiram Serra). These have variously involved analyzing the stability and purity of pharmaceutical preparations, in particular:
  - a. the impurity of stock preparations
  - b. the stability of nitroglycerine tablets to heat;
  - c. the stability over several months of methyl-dopa, prednisone and banthine when these compounds were formulated into syrups.

- 3) Drug Assay Laboratory Department of Pharmacology, Stanford University: Research personnel from this laboratory (Director: Summer M. Kalman) have requested mass spectra on various derivatives of digoxin using both high and low resolution data.
- 4) Department of Psychiatry, Stanford University: The research group headed by Dr. J. Barchas has used low resolution mass spectral data for the purpose of structure elucidation of a basic compound of interest to their research program.
- 5) Department of Anesthesia, Stanford University: The DENDRAL group was asked by Dr. J. Trudell to help him in the identification of a urinary metabolite isolated after the administration of an anesthetic. This work involved high resolution mass spectrometry of fractions isolated by Dr. Trudell.
- 6) Department of Psychiatry, Palo Alto Veterans Hospital: In this work we analyzed samples by GC/MS given to us by Dr. S. Kanter who works with Dr. Hollister. They were interested in detecting cannabinal, delta-9-tetrahydrocannabinol and an unknown (molecular weight 312) from urine extracts of subjects who had smoked marijuana. This involved running standards of cannabinal and its delta-9-tetrahydro analog through the GC/MS. We were unable to identify these compounds by mass spectrometry as being present in urine. In a subsequent meeting we learned that their concentration was less than 20 nanogram (per GC/MS injection) which is below the limits of sample flow for the recording of reproducible mass spectra. Dr. Kanter is working on the problem of isolating sufficient material for GC/MS and we expect to continue this project in year II of the current grant.
- 7) Prof. McCarty - Civil Engineering: Prof. McCarty is involved in a project to monitor water quality of effluents from tertiary sewage plants. This project includes significant efforts at characterization of the organic content of the water in various phases of its treatment to determine the efficiency of removal of various materials and to identify unknown organic compounds. We have agreed to provide instrumental and computer program support where necessary to assist him in characterization of these samples.

#### Applications by Non-Stanford Scientists

As an additional component of the resource sharing aspects of research, we have, as resources allow, extended the use of our facilities to a group of users remote from the local Stanford community. We have divided these users into two categories, those for whom we have provided mass spectrometry support and those who represent users of DENDRAL programs and collaborators on program development via the SUMEX resources.

#### A. Users of Mass Spectrometry Facilities

- 1) Professor O. O. Orazi, La Plata, Argentina: During the past year we have supplied Dr. Orazi with three low resolution mass spectra. We will be providing HRMS data for him in year II of our grant.
- 2) Professor T. Nakano, Caracas, Venezuela: Dr. Nakano sent one sample of an unknown alkaloid for high resolution mass spectrometry. We were able to show that his low resolution mass spectrum was 2 amu from the true molecular ion and after recording a low resolution mass spectrum his alkaloid was identified as a known compound.
- 3) Dr. Steen Hammerum, Copenhagen, Denmark: Dr. Hammerum requested our assistance in running ultra high resolution mass measurements on several ions in the mass spectra of compounds he had specifically labelled with  $^{13}\text{C}$ .

#### B. Users/Collaborators of/with DENDRAL Programs on SUMEX

Below, in alphabetical order, we list those persons who have a) expressed interest in use of our programs and have been sent instructions in how to gain access to SUMEX and our programs. In many cases these persons have received more detailed information in the form of demonstrations in person or remotely using the LINK facilities of SUMEX, and b) persons who have acted as collaborators in development of parts of one or more of our programs.

Because we have just begun encouraging a significant community of persons to try our programs, we do not yet have a good idea of which persons will continue as serious users. But we have at least provided the opportunity for persons to gain access to our programs, try them and determine how they might (or might not) fit into their own research problems. The term "exploratory" refers precisely to this category of persons who are now engaged in this kind of evaluation. The program names after each person's activity refer to their current major interest. In some cases, we do not actually know the specific problems which are being explored.

1. Dr. A.L. Burlingame (U.C. Berkeley) - Exploratory - all DENDRAL programs.
2. Prof. E.J. Corey (Harvard) - Exploratory, collaboration on programming strategies, CONGEN.
3. Dr. L. Dunham (Zoecon) - Exploratory - Structure determination - CONGEN.
4. Dr. H.M. Fales (NIH) - Exploratory - all DENDRAL programs.
5. R. Feldmann (NIH) - Collaborative development of programs (structure input and drawing routines).
6. Prof. D.L. Fishel (Kent State) - Considering access to SUMEX - has the program descriptions.

7. Prof. M.J. Goldstein (Cornell) - We have provided CONGEN results to him for a difficult structure problem.
8. Dr. N.A.B. Gray (Cambridge) - Collaborating on strategies for computer-assisted structure elucidation programs. He is working on spectral data interpretation.
9. Dr. P. Gund (Merck, Sharpe & Dohme) - Arranging an on-line demonstration for exploratory purposes - CONGEN.
10. Dr. J. Karliner (Ciba-Geigy) - Using CONGEN on structure elucidation problems.
11. Dr. S. Heller (Environmental Protection Agency) - Collaboration on mass spectral library development.
12. Dr. P. Jurs (Penn. State) - Collaboration on structure analysis and building of chemical structure models.
13. Dr. B. Kowalski (Univ. of Washington) - Has approached us for use of SUMEX in pattern recognition work.
14. Dr. D. Lefkowitz (Univ. of Penn.) - Exploratory - interest in DRAW portion of CONGEN for NCI chemical information system.
15. Dr. S. Markey (NIH) - Exploratory - all DENDRAL programs.
16. Dr. F. McLafferty (Cornell) - Exploratory - all DENDRAL Programs.
17. Dr. R. Milberg (National Center for Tox. Res.) - Exploratory - CONGEN.
18. Dr. D. Poulter (Univ. of Utah) - Exploratory - use of CONGEN in structure determination problems, especially terpenoids.
19. Dr. K. Rinehart (Univ. of Illinois) - Exploratory - all DENDRAL programs.
20. Dr. P. Roller (National Cancer Institute) - Exploratory - all DENDRAL programs.
21. Dr. R. Rosen (FMC Corp.) - Exploratory - all DENDRAL programs.
22. Dr. G. Szonyi (Polaroid Corp.) - Interest in CONGEN, exploratory phase beginning.
23. Dr. W.T. Wipke (Princeton) - Exploratory - CONGEN collaboration on structure model building and methods for stereochemical representation of chemical structure.

For a further discussion of the efforts by the DENDRAL project toward network collaboration and dissemination of the programs, see Appendix F which contains a preprint of a paper to be presented at the American Chemical Society symposium on Computer Networking in Chemistry in August of 1975.

#### DENDRAL PUBLICATIONS

- (1) J. Lederberg, "DENDRAL-64 - A System for Computer Construction, Enumeration and Notation of Organic Molecules as Tree Structures and Cyclic Graphs", (technical reports to NASA, also available from the author and summarized in (12)). (1a) Part I. Notational algorithm for tree structures (1964) CR.57029 (1b) Part II. Topology of cyclic graphs (1965) CR.68898 (1c) Part III. Complete chemical graphs; embedding rings in trees (1969)
- (2) J. Lederberg, "Computation of Molecular Formulas for Mass Spectrometry", Holden-Day, Inc. (1964).
- (3) J. Lederberg, "Topological Mapping of Organic Molecules", Proc. Nat. Acad. Sci., 53:1, January 1965, pp. 134-139.
- (4) J. Lederberg, "Systematics of organic molecules, graph topology and Hamilton circuits. A general outline of the DENDRAL system." NASA CR-48899 (1965)
- (5) J. Lederberg, "Hamilton Circuits of Convex Trivalent Polyhedra (up to 18 vertices), Am. Math. Monthly, May 1967.
- (6) G. L. Sutherland, "DENDRAL - A Computer Program for Generating and Filtering Chemical Structures", Stanford Artificial Intelligence Project Memo No. 49, February 1967.
- (7) J. Lederberg and E. A. Feigenbaum, "Mechanization of Inductive Inference in Organic Chemistry", in B. Kleinmuntz (ed) Formal Representations for Human Judgment, (Wiley, 1968) (also Stanford Artificial Intelligence Project Memo No. 54, August 1967).
- (8) J. Lederberg, "Online computation of molecular formulas from mass number." NASA CR-94977 (1968)
- (9) E. A. Feigenbaum and B. G. Buchanan, "Heuristic DENDRAL: A Program for Generating Explanatory Hypotheses in Organic Chemistry", in Proceedings, Hawaii International Conference on System Sciences, B. K. Kinariwala and F. F. Kuo (eds), University of Hawaii Press, 1968.
- (10) B. G. Buchanan, G. L. Sutherland, and E. A. Feigenbaum, "Heuristic DENDRAL: A Program for Generating Explanatory Hypotheses in Organic Chemistry". In Machine Intelligence 4 (B.

- Meltzer and D. Michie, eds) Edinburgh University Press (1969), (also Stanford Artificial Intelligence Project Memo No. 62, July 1968).
- (11) E. A. Feigenbaum, "Artificial Intelligence: Themes in the Second Decade". In Final Supplement to Proceedings of the IFIP68 International Congress, Edinburgh, August 1968 (also Stanford Artificial Intelligence Project Memo No. 67, August 1968).
- (12) J. Lederberg, "Topology of Molecules", in The Mathematical Sciences - A Collection of Essays, (ed.) Committee on Support of Research in the Mathematical Sciences (COSRIMS), National Academy of Sciences - National Research Council, M.I.T. Press, (1969), pp. 37-51.
- (13) G. Sutherland, "Heuristic DENDRAL: A Family of LISP Programs", to appear in D. Bobrow (ed), LISP Applications (also Stanford Artificial Intelligence Project Memo No. 80, March 1969).
- (14) J. Lederberg, G. L. Sutherland, B. G. Buchanan, E. A. Feigenbaum, A. V. Robertson, A. M. Duffield, and C. Djerassi, "Applications of Artificial Intelligence for Chemical Inference I. The Number of Possible Organic Compounds: Acyclic Structures Containing C, H, O and N". Journal of the American Chemical Society, 91:11 (May 21, 1969).
- (15) A. M. Duffield, A. V. Robertson, C. Djerassi, B. G. Buchanan, G. L. Sutherland, E. A. Feigenbaum, and J. Lederberg, "Application of Artificial Intelligence for Chemical Inference II. Interpretation of Low Resolution Mass Spectra of Ketones". Journal of the American Chemical Society, 91:11 (May 21, 1969).
- (16) B. G. Buchanan, G. L. Sutherland, E. A. Feigenbaum, "Toward an Understanding of Information Processes of Scientific Inference in the Context of Organic Chemistry", in Machine Intelligence 5, (B. Meltzer and D. Michie, eds) Edinburgh University Press (1970), (also Stanford Artificial Intelligence Project Memo No. 99, September 1969).
- (17) J. Lederberg, G. L. Sutherland, B. G. Buchanan, and E. A. Feigenbaum, "A Heuristic Program for Solving a Scientific Inference Problem: Summary of Motivation and Implementation", Stanford Artificial Intelligence Project Memo No. 104, November 1969.
- (18) C. W. Churchman and B. G. Buchanan, "On the Design of Inductive Systems: Some Philosophical Problems". British Journal for the Philosophy of Science, 20 (1969), pp. 311-323.
- (19) G. Schroll, A. M. Duffield, C. Djerassi, B. G. Buchanan, G. L. Sutherland, E. A. Feigenbaum, and J. Lederberg, "Application of Artificial Intelligence for Chemical Inference III. Aliphatic Ethers Diagnosed by Their Low Resolution Mass Spectra and NMR Data". Journal of the American Chemical Society, 91:26 (December 17, 1969).



- (20) A. Buchs, A. M. Duffield, G. Schroll, C. Djerassi, A. B. Delfino, B. G. Buchanan, G. L. Sutherland, E. A. Feigenbaum, and J. Lederberg, "Applications of Artificial Intelligence For Chemical Inference. IV. Saturated Amines Diagnosed by Their Low Resolution Mass Spectra and Nuclear Magnetic Resonance Spectra", *Journal of the American Chemical Society*, 92, 6831 (1970).
- (21) Y.M. Sheikh, A. Buchs, A.B. Delfino, G. Schroll, A.M. Duffield, C. Djerassi, B.G. Buchanan, G.L. Sutherland, E.A. Feigenbaum and J. Lederberg, "Applications of Artificial Intelligence for Chemical Inference V. An Approach to the Computer Generation of Cyclic Structures. Differentiation Between All the Possible Isomeric Ketones of Composition C<sub>6</sub>H<sub>10</sub>O", *Organic Mass Spectrometry*, 4, 493 (1970).
- (22) A. Buchs, A.B. Delfino, A.M. Duffield, C. Djerassi, B.G. Buchanan, E.A. Feigenbaum and J. Lederberg, "Applications of Artificial Intelligence for Chemical Inference VI. Approach to a General Method of Interpreting Low Resolution Mass Spectra with a Computer", *Chem. Acta Helvetica*, 53, 1394 (1970).
- (23) E.A. Feigenbaum, B.G. Buchanan, and J. Lederberg, "On Generality and Problem Solving: A Case Study Using the DENDRAL Program". In *Machine Intelligence 6* (B. Meltzer and D. Michie, eds.) Edinburgh University Press (1971). (Also Stanford Artificial Intelligence Project Memo No. 131.)
- (24) A. Buchs, A.B. Delfino, C. Djerassi, A.M. Duffield, B.G. Buchanan, E.A. Feigenbaum, J. Lederberg, G. Schroll, and G.L. Sutherland, "The Application of Artificial Intelligence in the Interpretation of Low Resolution Mass Spectra", *Advances in Mass Spectrometry*, 5 (1971), 314.
- (25) B.G. Buchanan and J. Lederberg, "The Heuristic DENDRAL Program for Explaining Empirical Data". In proceedings of the IFIP Congress 71, Ljubljana, Yugoslavia (1971). (Also Stanford Artificial Intelligence Project Memo No. 141.)
- (26) B.G. Buchanan, E.A. Feigenbaum, and J. Lederberg, "A Heuristic Programming Study of Theory Formation in Science." In proceedings of the Second International Joint Conference on Artificial Intelligence, Imperial College, London (September, 1971). (Also Stanford Artificial Intelligence Project Memo No. 145.)
- (27) Buchanan, B. G., Duffield, A.M., Robertson, A.V., "An Application of Artificial Intelligence to the Interpretation of Mass Spectra", *Mass Spectrometry Techniques and Appliances*, Edited by George W. A. Milne, John Wiley & Sons, Inc., 1971, p. 121-77.
- (28) D.H. Smith, B.G. Buchanan, R.S. Engelmores, A.M. Duffield, A. Yeo, E.A. Feigenbaum, J. Lederberg, and C. Djerassi, "Applications of Artificial Intelligence for Chemical Inference VIII. An approach to the Computer Interpretation of the High Resolution Mass Spectra of Complex Molecules. Structure Elucidation of Estrogenic Steroids", *Journal of the American Chemical Society*, 94, 5962-5971 (1972).

- (29) B.G. Buchanan, E.A. Feigenbaum, and N.S. Sridharan, "Heuristic Theory Formation: Data Interpretation and Rule Formation". In Machine Intelligence 7, Edinburgh University Press (1972).
- (30) Lederberg, J., "Rapid Calculation of Molecular Formulas from Mass Values". Jnl. of Chemical Education, 49, 613 (1972).
- (31) Brown, H., Masinter L., Hjelmeland, L., "Constructive Graph Labeling Using Double Cosets". Discrete Mathematics, 7 (1974), 1-30. (Also Computer Science Memo 318, 1972).
- (32) B. G. Buchanan, Review of Hubert Dreyfus' "What Computers Can't Do: A Critique of Artificial Reason", Computing Reviews (January, 1973). (Also Stanford Artificial Intelligence Project Memo No. 181)
- (33) D. H. Smith, B. G. Buchanan, R. S. Engelmores, H. Adlerkreutz and C. Djerassi, "Applications of Artificial Intelligence for Chemical Inference IX. Analysis of Mixtures Without Prior Separation as Illustrated for Estrogens". Journal of the American Chemical Society 95, 6078 (1973).
- (34) D. H. Smith, B. G. Buchanan, W. C. White, E. A. Feigenbaum, C. Djerassi and J. Lederberg, "Applications of Artificial Intelligence for Chemical Inference X. Intsum. A Data Interpretation Program as Applied to the Collected Mass Spectra of Estrogenic Steroids". Tetrahedron, 29, 3117 (1973).
- (35) B. G. Buchanan and N. S. Sridharan, "Rule Formation on Non-Homogeneous Classes of Objects". In proceedings of the Third International Joint Conference on Artificial Intelligence (Stanford, California, August, 1973). (Also Stanford Artificial Intelligence Project Memo No. 215.)
- (36) D. Michie and B.G. Buchanan, "Current Status of the Heuristic DENDRAL Program for Applying Artificial Intelligence to the Interpretation of Mass Spectra". August, 1973. To appear in Computers for Spectroscopy (ed. R.A.G. Carrington) London: Adam Hilger. Also: University of Edinburgh, School of Artificial Intelligence, Experimental Programming Report No. 32 (1973).
- (37) H. Brown and L. Masinter, "An Algorithm for the Construction of the Graphs of Organic Molecules", Discrete Mathematics (in press). (Also Stanford Computer Science Dept. Memo STAN-CS-73-361, May, 1973)
- (38) D.H. Smith, L.M. Masinter and N.S. Sridharan, "Heuristic DENDRAL: Analysis of Molecular Structure," Proceedings of the NATO/CNNA Advanced Study Institute on Computer Representation and Manipulation of Chemical Information (W. T. Wipke, S. Heller, R. Feldmann and E. Hyde, eds.) John Wiley and Sons, Inc., 1974.
- (39) R. Carhart and C. Djerassi, "Applications of Artificial Intelligence for Chemical Inference XI: The Analysis of C13 NMR Data for Structure Elucidation of Acyclic Amines", J. Chem. Soc. (Perkin II), 1753 (1973).

- (40) L. Masinter, N.S. Sridharan, R. Carhart and D.H. Smith,  
"Application of Artificial Intelligence for Chemical Inference  
XII: Exhaustive Generation of Cyclic and Acyclic Isomers".  
Journal of the American Chemical Society, 96 (1974), 7702. (Also  
Stanford Artificial Intelligence Project Memo No. 216.)
- (41) L. Masinter, N.S. Sridharan, R. Carhart and D.H. Smith,  
"Applications of Artificial Intelligence for Chemical Inference  
XIII: An Algorithm for Labelling Chemical Graphs". Journal of the  
American Chemical Society, 96 (1974), 7714.
- (42) N.S. Sridharan, Computer Generation of Vertex Graphs, Stanford CS  
Memo STAN-CS-73-381, July, 1973.
- (43) N.S. Sridharan, et.al., A Heuristic Program to Discover Syntheses  
for Complex Organic Molecules, Stanford CS Memo STAN-CS-73-370,  
June, 1973. (Also Stanford Artificial Intelligence Project Memo  
No. 205.)
- (44) N.S. Sridharan, Search Strategies for the Task of Organic  
Chemical Synthesis, Stanford CS Memo STAN-CS-73-391, October,  
1973. (Also Stanford Artificial Intelligence Project Memo No.  
217.)
- (45) D. H. Smith, "Applications of Artificial Intelligence for  
Chemical Inference XIV: The Number of Structural Isomers of  
 $C^xN^yO^z$ ,  $x + y + z \leq 6$ . An Investigation of Some Intuitions of  
Chemists."
- (46) D. H. Smith, "Applications of Artificial Intelligence for  
Chemical Inference XV." In preparation.
- (47) D. H. Smith and R. E. Carhart, "Applications of Artificial  
Intelligence for Chemical Inference XVI: On Structural Isomerism  
of Tricyclodecanes." To be submitted to Journal of the American  
Chemical Society.
- (48) R. G. Dromey, B. G. Buchanan, D. H. Smith, J. Lederberg and C.  
Djerassi, "Applications of Artificial Intelligence for Chemical  
Inference XVII: A General Method for Predicting Molecular Ions in  
Mass Spectra." To appear in Journal of Organic Chemistry, March,  
1975.
- (49) B. G. Buchanan, "Scientific Theory Formation by Computer." To  
appear in Proceedings of NATO Advanced Study Institute on Computer  
Oriented Learning Processes, 1974, Bonas, France.
- (50) E. A. Feigenbaum, "Computer Applications: Introductory Remarks,"  
in Proceedings of Federation of American Societies for  
Experimental Biology, Vo. 33, No. 12 (Dec., 1974) 2331-2332.

## IV.A.1.b MYCIN PROJECT

## Computer Based Consultation in Clinical Therapeutics

Prof. S. Cohen, M.D. (Pharmacology) and  
Dr. B. Buchanan (Computer Science)

(Grant HEW HSO-1544-01, 3 years, \$116,734 this year)

The overall objective of the MYCIN project at Stanford is the development of a computer-based system which is capable of using both clinical data and the judgmental knowledge of experts to improve the effectiveness of medical decision making with regard to clinical therapeutics. The work concentrates initially on the use of antimicrobial agents in the treatment of bacteremias.

The present state of the research includes the following accomplishments:

- 1) An interactive program which utilizes data available from the microbiology and clinical chemistry laboratories, plus the physician's response to computer-generated questions, to provide physicians with consultative advice on antimicrobial therapy. Therapy selection takes into account both the patient's infections and the identities of the organisms causing these infections.
- 2) An interactive explanation capability to permit the program to explain all of its actions and reasoning, including, for example, the deduction of the identities of pathologic agents.
- 3) A capability for computer acquisition of judgmental knowledge about those concepts which the program uses in making deductions; this permits experts in the field of infectious disease therapy to teach the system those therapeutic decisions which they find useful in their clinical practices.

Goals for further development of the system include:

- 1) Expansion of the consultation program to deal with infections other than bacteremias.
- 2) Implementation of the system in the clinical setting at the Stanford Hospital.
- 3) Evaluation of the clinical usefulness of the system and of its impact upon the clinical staff and their prescribing habits.
- 4) Expansion of the rule acquisition system to allow experts to introduce new concepts and decision rules which use these concepts.

- 5) Integration of the rule-acquisition system and explanation capabilities to allow experts to enter new decision rules dynamically during a consultation. This will be useful when the program reaches a conclusion which the expert believes to be erroneous, and the explanation facility indicates that it was reached because the system lacked a vital concept or decision rule.
- 6) The development of meta-rules which will be used for expressing strategies for approaching clinical problems.

The techniques for acquisition, representation, and utilization of knowledge, plus considerations of natural language processing, draw upon current research in artificial intelligence.

#### Past Year's Accomplishments

In the past year we have sought to (i) improve the validity of the system's therapeutic advice and increase the scope of its competence; (ii) redesign the control structure to provide a faster, more direct and more general implementation, with increased attention to human engineering; (iii) develop the program's ability to explain its actions to the user; (iv) develop the capability to acquire new rules through interactions with experts, and (v) evaluate the program's competence and performance in a clinical setting.

#### Validity and Scope

Although MYCIN's original focus was directed only at patients with positive blood cultures, the basic methodology was intended to support a much more general approach to the problem. In the past year the system has gained the ability to deal with infections from which the causative pathogen hasn't been isolated (e.g. pneumonia), or which haven't even been cultured (e.g. brain abscess). In broadening the program from bacteremia, we have also acquired the ability to evaluate the meaningfulness of isolates. In addition, the program has been given a sense of time in order to cope with more easily with the order of events in the patient's history, such as a sequence of cultures. This has made its reasoning more powerful by eliminating, for example, the need to ask explicitly about the order of every pair of events.

An extensive review of the program's approach to drug selection has led to a major revision in the basis for therapy selection. The program now deduces the identity of the organism, the infectious disease diagnosis, and the significance of the organism. These are the primary factors in drug selection, with drug toxicity and ecological factors as secondary considerations. The result is a more appropriate, more sharply focussed drug selection that is able to specify dose, route, and duration.

A review of the treatment of various medical concepts has provided a far more comprehensive collection of rules, which now number almost 200. We have revised the program's approach to some concepts, and provided a much more consistent level of knowledge by filling in many gaps. As part of our efforts to introduce the system to clinicians, it was presented at the meeting of the American Federation of Clinical Research in February, 1975 [1].

### Generality and Usability

A comprehensive review and modification of the control structure was undertaken to improve the program's efficiency and generality. The resulting program is smaller, faster, more direct, and yet more general than the original; response times are now no more than 15 seconds, and typically much less, core size is down by 15%. Also, several new capabilities were added to make the program easier to use. The system is now more tolerant of erroneous or inappropriate responses, and can provide a reworded question, along with a list of acceptable answers. In addition, it can recognize responses which are not sufficiently precise, and rephrase its questions accordingly. (A review of design considerations for the system as a whole is in [2]).

### Explanations

We have developed the explanation facility both to insure that the user can understand the rationale behind the system's recommendations, and to educate him, as well. The system currently can explain both the motivation for questions it asks and the source of the conclusions it draws, as well as answer general questions concerning its store of medical knowledge. All of these capabilities were augmented during the past year, some extensively.

The explanation of motivations for questions underwent a major design revision, which resulted in a much more powerful approach based on the program's knowledge of its own control structure and its ability to examine its rules. The user can now fully explore the system's current line of reasoning.

The language understanding capabilities of the question answering system have also been revised, allowing a broader range of questions to be asked and offering more precise answers. The use of this feature has also been simplified somewhat, so that the user no longer needs to classify all his questions.

### Knowledge Acquisition

A preliminary knowledge acquisition system was completed in the middle of the year. We have demonstrated that a physician can teach the system new rules in a rather stylized subset of English. (An overview of the capabilities is in [3]). Building on the experience gained here, a redesigned system is currently being constructed, which allows the user to examine and modify the program's knowledge and

behavior as a single, unified action. That is, the functions of two separate modules have been combined in a single, redesigned system, that will make use of the fact that the nature of the explanations requested can give a clear hint as to the content of the new rule. It will also advise the user as to the effect of his rule on the original deficiency -- i.e. whether or not it corrects the problem he noticed. Progress has also been made on the issue of assuring the consistency of the knowledge base by examining the nature of contradictions as they appear in our multi-valued logic system.

## Evaluation

Much of the focus for the development in the past year was suggested by the results of a preliminary evaluation done in May, 1974 [4]. This concentrated on the validity and acceptability of the program's advice, by comparing its performance on fifteen cases to that of five infectious disease experts reviewing the same cases. Even at this early stage of development, the results were quite encouraging. None of questions generated by the program were judged irrelevant by a majority of the physicians. In its final diagnosis, the program offered an average of 4.0 possible identities for each organism, while the experts suggested an average of 5.3 possibilities. In almost 50% of the cases, the program's therapeutic regimen was identical to that suggested by the physicians, and 72% of the time it was judged an acceptable alternative. (Agreement among experts was about the same as agreement with the program.)

Another, much more comprehensive study is just beginning, with the primary focus on the validity of the program's medical advice, its impact on the prescribing habits of the user community, and its acceptance by clinicians, as indicated by their use of the system.

## Comments on SUMEX

One of the most important elements in our attempt to develop a truly competent program is the repeated testing of performance on a wide range of cases. The shared resource concept of SUMEX, with the availability of the system to numerous research groups, has become a significant aid in this testing process. We have begun to receive useful feedback from groups at the University of Rochester and the University of Virginia, both of which have been using the program.

The SNDMSG and other system communication facilities have made communication simple, direct, and effective, despite the large distances involved.

We anticipate that, with the availability of a new and redesigned explanation and knowledge acquisition system (planned for mid-summer), we can widen the scope of this interaction. Rather than having users report bugs to us, requesting information, and our answering their questions and fixing the problems, the system itself

should be capable of handling a subset of such interactions. This will help to put the user in direct touch with the knowledge base of the program, so that he can modify or augment it directly. Thus, in addition to gathering the usual sort of feedback from user experience in running the program, we hope to benefit directly from the expertise of infectious disease experts in various centers across the country.

#### References

- [1] E H Shortliffe, F Rhame, et. al., "MYCIN, A Computer Program providing Antimicrobial Therapy Recommendations", Clinical Research, vol 23, p 107A (abstract) 1975.
- [2] E. H. Shortliffe, S. G. Axline, B. G. Buchanan, S. N. Cohen, "Design Considerations for a Program to Provide Consultations in Clinical Therapeutics". Presented at San Diego Biomedical Symposium 1974 (February 6-8, 1974).
- [3] E. H. Shortliffe, R. Davis, S. G. Axline, B. G. Buchanan, C. C. Green and S. N. Cohen, "Computer-Based Consultations in Clinical Therapeutics: Explanation and Rule Acquisition Capabilities of the MYCIN System", to appear in Computers and Biomedical Research, June 1975
- [4] E H Shortliffe, "MYCIN, A Rule Based Computer Program..." STAN-CS-74-465 Computer Science Department, Stanford University 1974
- [5] E. H. Shortliffe, S. G. Axline, B. G. Buchanan, T. C. Merigan, and S. N. Cohen, "An Artificial Intelligence Program to Advise Physicians Regarding Antimicrobial Therapy." Computers and Biomedical Research, 6 (1973), 544-560.
- [6] E. H. Shortliffe and B. G. Buchanan, "A Model of Inexact Reasoning in Medicine, July, 1974. To appear in Mathematical Biosciences.



## IV.A.1.c PROTEIN STRUCTURE MODELING PROJECT

## Protein Crystallography Project

Dr. S. Freer (Chemistry, U. C. San Diego) and  
Prof. E. Feigenbaum and Dr. R. Engelman (Comp. Sci., Stanford)

(Grant NSF DCR74-23461, 2 years, \$88,436 total)

## I. General Objectives

The protein crystallography project involves scientists at two different universities, pooling their respective talents in protein crystallography and computer science, and using the SUMEX-AIM facility as the central repository for programs, data and other information of common interest. The two groups involved are members of the Computer Science Department at Stanford (Prof. Edward Feigenbaum and Dr. Robert Engelman) and members of the Department of Chemistry at the University of California at San Diego (Prof. Joseph Kraut and Dr. Stephan Freer). The general objective of the project is to apply problem solving techniques, which have emerged from artificial intelligence research, to the well known "phase problem" of x-ray crystallography, in order to determine the three-dimensional structures of proteins. The work is intended to be of practical as well as theoretical value to both computer science (particular AI research) and protein crystallography. Viewed as an AI problem, the objectives are:

1. To discover from expert protein crystallographers the knowledge and heuristics which could be used to infer the tertiary structure of proteins from x-ray crystallographic data, and to formalize this expertise in appropriate data structures and heuristic procedures.
2. To discover a program organization and a set of representations which will allow the knowledge and the heuristics to cooperate in making the search efficient, i.e., generating plausible candidates in a reasonable time.
3. To implement the above in a system of computer programs, the competence of which will have a noticeable impact upon the discipline of protein crystallography.

As a research task in protein crystallography, the objective is to develop a computational system which can infer the tertiary structure of a protein molecule in the absence of phase information normally obtained from multiple isomorphous replacement procedures.

## II. Specific Objectives

We are attacking the phase problem in protein crystallography by

developing a system of computer programs that will enable us to make progress in structure determination without use of multiple isomorphous replacements. Protein crystal structures can be divided into three classes according to how much of the structure is known at the outset of the investigation. In order of increasing difficulty these are:

- I. three-dimensional structure already known, but from crystals in a different space group;
- II. three-dimensional structure of a homologous protein is known;
- III. only the amino acid sequence is known or can be approximately inferred from homologous proteins.

We have selected four proteins for study, which fall into the three classes as shown below:

I	II	III
Structure known	Homologous Structure Known	Amino Acid sequence only
Chymotrypsinogen (type D crystals)	Cytochrome f from Spirula	Two-iron ferredoxin from Anacystis nidulans
	Cytochrome c2 from rhps. capsulata	

For proteins in classes I and II we have used or plan to use Patterson search techniques and the rotation function of Rossmann and Blow. Our objective here is to directly compare the relative efficacy of the Patterson search and rotation function methods. The output from either of these methods is the orientation and location of a trial structure derived from the crystal structure of the identical or homologous protein (say the backbone atoms plus the beta-carbon atoms) in the unit cell of the unknown crystal structure. The correctness of the trial structure can be verified by crystallographic refinement (Freer, Alden, Carter and Kraut, J. Biol. Chem., v. 249, pp. - ). In order to make refinement a more useful tool for this verification we will introduce techniques for incorporating the amorphous solvent into the trial structure, in hopes of speeding the convergence of the refinement process.

The more difficult class III structure determination of the two-iron ferredoxin protein will be attacked by parallel application of Patterson search techniques and Fourier Bootstrap methods. In the Patterson search investigation we will first try to locate the iron atoms with an anomalous dispersion Patterson and then attempt to determine the orientation of the entire iron-sulfur group by Patterson search in which the iron-sulfur portion of the  $[\text{FeS}(\text{SCH}_2)_2\text{C}_6\text{H}_4]_2(2-)$  synthetic analogue described by Mayerle, et al. (Proc. Nat. Acad.

Sci., v. 79, p. 2429 (1973)) is used as the search group. In addition we will determine from known sequence structure correlations what if any helical regions are strongly predicted by the amino acid sequence and will then search the Patterson with Fe-helix vectors to confirm their existence and determine their position and orientation. We will continue this approach to build up as much of the structure as possible.

The Fourier Bootstrap method begins as a trial and error search in which a sphere is used as the trial structure (Kraut, Biochem. Biophys. Acta, v. 30, p. 264 (1958)). Classical Fourier refinement will be used to reveal the shape of the protein and locate its position with more precision. The refinement will then proceed by direct modification of the electron-density map (eliminating regions of negative density, etc.) followed by Fourier inversion of the modified map, a technique used at UCSD from time to time during the last ten years and which has been employed recently with success by Collins and Legg (Amer. Cryst. Assoc., program and abstracts of summer meeting, v. 261 (1974)) to refine the phases and extend the resolution of the rubredoxin crystal structure.

### III. Summary of Project Accomplishments

Our efforts during the past year have been mainly in building the tools which will be the "expert" components in our overall structure elucidation system. The main accomplishments are listed below.

1. Conversion of the Patterson Search program, PSRCH, to SUMEX. This program is now operational.
2. Development of the Superposition program to infer positions of additional atoms when the coordinates of a partial structure are known. This program is now operational.
3. Conversion of the Rossmann rotation function program to SUMEX. This program was obtained from Rossmann's laboratory at Purdue University. Although the conversion to FORTRAN-10 was straightforward, it has taken several months, even with help from one of the program's authors, to learn to obtain good results from it on test problems. It now performs in some cases at a level similar to the Patterson search program. However, attempts to find non-crystallographic symmetries in real crystal protein structures have thus far been unsuccessful. Testing and debugging of the program is still in progress. Unfortunately, the crystallography group at Purdue has no routine access to computing terminals, so that real-time consulting (see below) has not been exploited.
4. IDATA2, a program for computing theoretical structure factors, given the atomic coordinates of a complete molecule, was transferred from UCSD, where it operates on a CDC-3600, to SUMEX and made operational on SUMEX. As originally received, IDATA2 was

a non-standard Fortran program, using many subroutines written expressly for the CDC-3600 either in a CDC dialect of Fortran or in assembly language. Conversion of the program to SUMEX was considerably expedited by the existence of the TYMNET connection to UCSD, and the linking facility in TENEX. Dr. Stephan Freer, one of the authors of IDATA2, spent several hours at a terminal in his office at UCSD, logged in to SUMEX and linked to Dr. Robert Engelmores at Stanford, while Engelmores went through many iterations of editing, compiling, and (eventually) executing the program. Freer's real-time consulting (a further example of which is given in the next section) was as valuable as if he were working in the same office as Engelmores, and compressed the program conversion time from several weeks to several days.

5. Several ancillary utility programs for translating and rotating sets of atom coordinates, determining structural parameters, reading and writing files containing structure factors, Patterson tables, etc. have been implemented on SUMEX.

#### IV. Interaction with the SUMEX resource

This project was conceived from the outset with the expectation of using a common computing facility, available to one or both groups via remote terminals connected to the facility over a high speed network, such as the ARPANET. Preliminary work began before such a facility existed, and it was necessary to make frequent and lengthy visits to UCSD to obtain the necessary expertise and guidance. Moreover, the first computer programs had to run on two very different computers (a CDC 3600 and an IBM 360/67) so that program development could continue at either university. Most of the time interaction between the two research groups consisted of occasional phone calls or letters. There was typically a turn-around time of several days between a suggestion for debugging a program, or a formulation of a series of computer runs, and the corresponding results. By that time the person who had suggested the work originally had forgotten why, and needed to be brought up to speed again.

All that is past history. All program development, and most communications are now effected on the SUMEX computer. The UCSD group has a direct connection to SUMEX via the TYMNET and ARPANET computing networks. Routine daily communications now take place using the system's message facility. Program files are equally accessible from Stanford and UCSD, so that either group can construct, edit or exercise the programs. Large data files can be transmitted to and from either site via the ARPANET, obviating the ubiquitous problems of machine incompatibilities, difficulties of reading tapes written at foreign sites, etc.

But perhaps the most noticeable change in our modus operandi that has occurred with the advent of computer networking has been the closer coupling between the Stanford and UCSD groups in all phases of the research activity. The following transcript (edited for brevity) of a terminal session illustrates how the system is now used to provide "real time" consultation. A member of the Stanford group

(Engelmore) has been debugging one of the basic analytical programs, the Rossmann rotation function, and is seeking further help from one of the protein crystallographers (Freer) at UCSD, who is logged in to SUMEX and has linked his terminal to Engelmore's. The transcript begins shortly after both parties have been notified that their terminals are linked. (Freer's comments are in lower case, Engelmore's in upper case preceded by a semi-colon.)

LINK FROM FREER, TTY 77

@;HI, STEVE. WHAT'S UP?

have you tried a general rotation (about a non-crystallographic axis)?

@;NO, I WANT TO DO THAT NOW, AND NEED SOME ADVICE ABOUT SETTING  
@;IT UP. I WAS THINKING OF TAKING THAT LITTLE DI-PEPTIDE,  
@;ROTATING IT ABOUT SOME AXIS BY SOME ANGLE, ADDING THE NEW SET  
@;OF COORDINATES TO THE OLD SET, ESSENTIALLY MAKING A NEW  
@;MOLECULE WITH TWICE AS MANY ATOMS, AND COMPUTING A NEW FC FROM  
@;THERE.

all right, that is a good idea. it will check the capability of the program to locate non-crystallographic symmetry axes. remember to increase the size of the unit cell to accommodate the additional atoms.

@;LET ME SHOW YOU SOME OF THE COORDINATES AND THEN ASK A  
@;QUESTION.  
@TY THREON.ATM

; <ENGELMORE>THREON.ATM;3 THU 13-MAR-75 11:53AM PAGE 1

TITLE L-THREONYL-L-PHENYLALANINE-P-NITROBENZYL ESTER							
HYDROBROMIDE							
CELL	THRE	8.93	45.75	5.05	90.	90.	90.
GRID		10	10	10			
ATOMG	BR	4.563	0.363	-2.124	3.8	BR	
ATOMG	O	-1.867	0.274	-2.971	6.6	O1	
ATOMG	O	1.609	0.342	2.928	4.2	O2	
ATOMG	O	0.867	1.138	6.459	5.2	O3	

@;THERE ARE 30 ATOMS IN ALL. FIRST, SHOULD I DOUBLE THE CELL  
@;SIZE?

i think you will be ok if you must double the size of the cell.

@;DO YOU HAVE ANY SUGGESTIONS FOR THE DIRECTION OF THE ROTATION  
@;AXIS ABOUT WHICH I SHOULD ROTATE 180 DEGREES?

sorry for the delay, i am doing a couple of other things at the same time. you are more up on general rotations than i. you could make it parallel to a crystallographic axis if you wish (for ease of computation). just don't let it coincide with a

crystallographic axis.

@;OK, I THINK I WILL FIRST TRANSLATE THE MOLECULE BY SOME  
@;"RANDOM" AMOUNT IN THE X-Z PLANE, THEN ROTATE IT 180 DEGREES.  
@;OR VICE VERSA.

sounds ok.

are you going to perform the general rotation and increase the  
cell size and then run a self-rotation rotation function run  
while i am on line?

@;YES, LET'S DO IT. FIRST I'LL FIX UP THE CELL COORDINATES IN  
@;THAT FILE I JUST LISTED. YOU CAN JUST WATCH FROM HERE ON ,  
@;BUT FEEL FREE TO INTERRUPT AT ANY TIME.

(E. SETS UP AND RUNS A UTILITY PROGRAM TO GENERATE NEW ATOM  
COORDINATES. THE OUTPUT, WHICH IS PRINTED ON BOTH E'S AND F'S  
TERMINALS, IS OMITTED HERE FOR THE SAKE OF BREVITY.)

@;OK, NOW I HAVE TWO FILES OF ATOM COORDINATES TO MERGE. I DON'T  
@;HAVE ANY AUTOMATIC WAY TO SEE IF THEY'RE OVERLAPPING, SO I  
@;GUESS I'LL HAVE TO INSPECT THEM VISUALLY.

i wouldn't worry overly much about it. you have increased the  
volume of the unit cell by a factor of 8 and the contents by 2.  
a simple way to check is to sort on xyz and examine delta xyz of  
adjacent pairs and list the pair with the smallest delta. don't  
worry about it now, let's just go on and run the rot. program.

@;OK, WILL DO.

(E. SETS UP INPUT FOR THE PROGRAM WHICH COMPUTES THEORETICAL  
STRUCTURE FACTORS.)

@;STEVE, SHOULD I GENERATE THE FC IN P1 INSTEAD OF P2(1)2(1)2(1)?

no, you have things set up for p2(1)2(1)2(1) now, so just stay in  
that space group if you wish. the overlap problem would be  
completely avoided if you switched to p1, but i don't think it  
is necessary to do that now.

@IDATA2

(E. RUNS THE IDATA2 PROGRAM TO GENERATE STRUCTURE FACTORS. WHILE  
THE PROGRAM IS RUNNING AND PRINTING INTERMEDIATE RESULTS ON BOTH  
TERMINALS, AN ERROR IS DETECTED BY THE PROGRAM:)

TOO MANY REFLECTIONS

END OF EXECUTION

CPU TIME: 56.22 ELAPSED TIME: 3:2.91

EXIT.

@;TOO MANY REFLECTIONS????

remember, you doubled each cell edge and hence increased the volume of the cell by 8. you will therefore get 8 times as many reflections for a given minimum bragg spacing.

@;SO I GUESS I SHOULD ALSO REDUCE THE RESOLUTION FROM 2.0 TO 4.0,  
@;RIGHT?

yes, that will reduce the number of reflections by a factor of 8.

@;OK, HERE WE GO AGAIN.

#### V. Proposed Extensions to Scope of Project vis-a-vis SUMEX Use.

Some of the components of the structure elucidation system under construction here are standard crystallographic analysis programs, used routinely by the UCSD protein crystallographers. These programs were originally written for the CDC 3600 at UCSD. Although that machine will continue to be their primary computing facility, the UCSD crystallographers have expressed an interest in performing some of their data reduction and other computational work on SUMEX, because of its larger core size and shorter turn-around times. The ARPANET connection now makes this activity more attractive, since relatively large files of data must be transferred between UCSD and SUMEX. This extension of the use of SUMEX for "standard" crystallographic computations would be beneficial to the Stanford side of the collaboration as well, by coupling the two groups more tightly through mutual development and use of software. As the UCSD crystallographers perform more of their computations at SUMEX, the locus of interest in crystallographic computing will shift towards remote computing via networks, providing a further demonstration of the utility of the SUMEX concept.

#### IV.A.2 NATIONAL USERS

##### IV.A.2.a DIALOG PROJECT

Principal Investigators: Dr. H. Pople and J. Myers, M.D.  
University of Pittsburgh

(Grant HEW MB-00144-01, 3 years, \$167,168 this year)

#### I. ABSTRACT

DIALOG is a computer-based system for general medical consultation that incorporates a hypothesis-formation model of diagnostic logic, and an extensive medical data base now encompassing approximately half of the major diseases of internal medicine. The system has been designed to deal with complex clinical problems, confounded by data produced by two or more distinct clinicopathological entities. In dealing with such cases, the DIALOG heuristic focuses successively on various aspects of a problem, disregarding, as it does so, findings that are irrelevant in each context. The system thereby exhibits diagnostic behavior comparable to the problem-oriented approach of the skilled clinician.

#### II. SUMMARY OF PROJECT ACTIVITIES

When the DIALOG project was moved to SUMEX-AIM last fall, our first priority was to convert the existing DIALOG program and data base from LISP 1.6 to INTERLISP. This was accomplished without great difficulty, and a working version of the system was operational on SUMEX-AIM by late November.

At about the same time, an interactive data-entry system was devised so as to enable expansion and refinement of the data base. As an expedient, because the planned volume of data could not be accommodated within the user address space of INTERLISP, this system was implemented as an assemblage of five interacting INTERLISP forks. While serving its intended purpose, which was to permit a twofold expansion of the data base, this system proved unwieldy and costly in its utilization of system resources.

Design of a successor system that could provide efficient access to the newly expanded data base was begun in January of this year. While continuing with INTERLISP as the host processor, we decided to structure the entire vocabulary and network of associations comprising the DIALOG data base in a set of disk files, with pages to be mapped into a resident core buffer on a demand basis. This design required that data management, core management, data entry and editing routines be written in basic assembly language; coding and checkout of these program modules is ongoing at the present time.

Plans for field test of the DIALOG system, outlined in our



original SUMEX-AIM proposal, have been held in abeyance pending completion of this system design and implementation effort. It is now projected that the testing and evaluation activities will commence in early fall of this year.

### III. INTERACTION WITH SUMEX-AIM

We have found SUMEX-AIM to be a very habitable environment. The computer resources, both hardware and software, amply serve the present needs of the project and despite lack of local TENEX expertise, we have been able to accomplish a number of sophisticated systems programming tasks, thanks to superb documentation and a most helpful SUMEX-AIM staff.

Our only concern is that the SUMEX-AIM resource may have become overloaded, as the afternoon response time has degraded badly during recent weeks. We would hope that the physicians who are invited to aid in the evaluation of DIALOG this fall might be encouraged to exercise the system at their convenience, rather than at the convenience of the system scheduler.

#### IV.A.2.b     DISTRIBUTED DATA BASES FOR CHRONIC DISEASES

##### Distributed Data Base System for Chronic Diseases

R. A. Nordyke, M.D.; F. Kuo, Ph.D.; C. Kulikowski, Ph.D.

(Grant application in preparation)

Beginning in March 1975, we have begun to develop prototype consultation programs in chronic diseases that are to be linked to a set of distributed data bases. The collaborating institutions are Pacific Health Research Institute (R. A. Nordyke), University of Hawaii ALOHA System (F. Kuo) and Rutgers University (C. Kulikowski).

The initial phase consists of establishing a data base on thyroid diseases using already accumulated data. This will serve as a prototype for those in other chronic diseases such as hypertension and diabetes.

In developing computer programs for consultation, decision support and data acquisition and analysis, we recognize that different levels of complexity are appropriate depending on the particular patient's characteristics, the health care provider's role, and the purposes and environment of the encounter. To achieve a multiplicity of goals and provide a variety of clinical functions at different levels of sophistication it is most appropriate and effective if we distribute our work between local processors and shared national resources. In this manner we can best use of the specialized capabilities of each type of computer system and data base.

To satisfy local needs and preferences, we intend to design data acquisition and medical decision protocols to be run on small local minicomputers. These local minicomputers are to be linked via computer-communication networks (ALOHA and ARPANET) to the SUMEX computer at Stanford. Input/output control programs will be designed to facilitate modular and standardized transmission of important data that enter in the chronic disease data base. The ALOHANET can make the consultation programs available via radio-links and satellite to other Hawaiian islands and remote areas of the Pacific basin, where nurse/paramedic protocols for the management of chronic diseases could have a significant effect on the quality of health care delivery.

One of the principal objectives of our proposed research program is the development of sophisticated computer programs based upon methods of artificial intelligence to aid in the diagnosis and treatment of specific chronic diseases. This aspect of our work will build on already established collaborations between the clinical group at the Pacific Health Research Institute/Straub Clinic and the computer science group on artificial intelligence in medicine (AIM) at Rutgers University. Work at Rutgers (in collaboration with the Mt. Sinai School of Medicine) has concentrated on another chronic disease - glaucoma. Thus, the proposed work in thyroid diseases represents a broadening of scope within a class of similar problems.

The systems currently under development possess unique logical and inferential capabilities for the explanation and justification of their decisions and strategies, based on their use of rational models of normal body function, disease processes, and patterns of health needs and health care. Because our models allow explanation of the process under study at many levels of complexity, they can generate information for many levels of users. For example, within a consultation program, diagnostic reasoning can be stressed for inexperienced practitioners, while prognosis and recommendations for therapy can be emphasized for those bearing greater responsibility for patient care.

The consultation programs within SUMEX will provide: multiple modules of disease, varying levels of resolution, different modes of interpretation (causal, logical, taxonomic, associative, probabilistic, etc.) facilities for explanation, instruction and querying of a data base of existing cases. A control program at the central resource (SUMEX) will receive requests from the local clinical control program and decide on the appropriate level and scope of the response. The local minicomputer will limit the type of information transferred to the resource machine to maintain security and confidentiality of the medical information. Local mass data will be on disc and on a large time-sharing computer, the BCC 500 at the University of Hawaii.

The data base for a specific chronic disease is best established at a clinical node where the appropriate knowledge and experience is available. Access to the data base by other investigators can be attained either through SUMEX (for AIM researchers) or directly through one of the computer networks..justify

## IV.A.2.c HIGHER MENTAL FUNCTIONS MODELING

HIGHER MENTAL FUNCTIONS MODELING (HMF)  
Project Summary - April, 1975Kenneth M. Colby, M.D.  
Professor of Psychiatry, UCLA(Grants NIH MH-06645-13, 3 years, \$170,000 this year  
and NIH MH-27132-01, 2 years, \$130,000 this year)

Only since SUMEX has come onto the ARPA network have I been able to work on the machine. Hence, there is not much to report.

## 1) GOALS

We plan to construct, test and validate an improved version of a computer simulation of paranoid processes. This model has clinical implications for the understanding, treatment and prevention of paranoid disorders.

## 2) ACTIVITIES

The interactive model is now running on SUMEX, and we are collecting data from the interviews to improve the model's language recognition capabilities.

Now that SUMEX can be reached from UCLA, we expect to do a great deal of work on the facility, particularly in debugging the improved version of the model. Also, we intend to conduct some experimental tests of the model in which experienced clinicians rate its behavior along specific dimensions.

No publications as yet.

## 3) SUMEX INTERACTION

I have a lot of trouble getting on the machine. The messages are "Timed Out", "Host Dead" and/or "Connection Open", but nothing else happens. Also, the machine seems to go down a lot during the late morning and early afternoon.

[NOTE: "These difficulties evidently reflect early problems with reliability of ARPANET connection software and hardware." J.L.]

## IV.A.2.d MEDICAL INFORMATION SYSTEMS LABORATORY

## Principal Investigators:

Dr. B. McCormick and M. Goldberg, M.D.  
(Univ. of Illinois at Chicago Circle)

(Grant HEW MB-00114-01, 2 years, \$380,619 this year)

## (1) OBJECTIVES

The Medical Information Systems Laboratory (MISL) pursues three major activities: clinical research and decision support; construction and modeling of a data base in ophthalmology; and network-compatible data base design. The priorities among these are such that the latter two activities are ancillary to the exploration of artificial intelligence techniques in clinical decision making. The resource utilizes the computer facilities of the University of Illinois and the SUMEX-AIM network, and provides the administrative structure for assembling the expertise of the Department of Information Engineering and the Department of Ophthalmology of the Illinois Eye and Ear Infirmary.

Clinical decision support activities are currently proceeding in the following specific areas:

- a. Disease process nets: the investigation, development, and testing of knowledge-based models of disease processes.
- b. Data understanding systems in clinical medicine: development of a computer laboratory health care resource for the capture and validation of data at its source. The idea is to augment data capture with a clinical decision support model of what to seek, what to expect, and how to interpret data sets.
- c. Synthesis of optimal protocols: development of a formal system, called a "variable-valued logic system," with a few simple operations which -- when supplied with the observed data sets and the relations between the observed variables -- produces a description which is minimal. The system will be used to deduce the simplest rule for recognizing one class in the limited context of another class (or classes) of objects, and so prescribe the protocol for a differential diagnosis.
- d. Physician-guided decision support: an effort to provide the clinician and clinical researcher with tools for validating his data base, comparing his decision-making protocols with those of his colleagues, and initiating epidemiological studies.
- e. Support for epidemiological and longitudinal studies: involves interactive graphics access to classical biostatistical support programs; analysis procedures as in automatic control theory; and

risk and health hazard appraisals on the basis of current epidemiological studies.

- f. Investigations into formal aspects of modeling and knowledge representation.
- g. Extensive study of existing systems for automated diagnosis.
- h. Creation of a natural language question-answering system for extracting information from disease model data bases.
- i. Sponsorship of series of seminars in relational data base design and applied epistemology.

## (2) PROGRESS

In MISL's first year most attention has been given to assembling necessary equipment, personnel, and ideas. SUMEX-AIM was used for the following purposes:

- a. to facilitate a substantial review of literature on disease modeling, formal modeling, and knowledge representation. Information obtained over SUMEX-AIM was a considerable help in assembling the relevant material.
- b. to communicate with other groups engaged in similar activities, especially the Computers in Biomedicine group at Rutgers.
- c. for the construction of INTERLISP-based graphical language and software for on-line graphics via a plasma panel.
- d. to edit and generate intra-project memos and reports using such SUMEX subsystems as SOS, TV-Edit, and PUB.

As MISL moves into its second year, its use of SUMEX-AIM is expected to increase substantially. Program development will proceed in the areas of disease modeling (of glaucoma, in conjunction with Rutgers; and diabetic retinopathy), natural language query systems, and systems for facilitating knowledge acquisition. SNOBOL and INTERLISP should receive very heavy use.

## (3) COMMENTS

Technically we have been well-served and well-pleased. The system has proven to be easy to use and immensely powerful. The services of Phil Jackson as consultant were very helpful in working out some knotty system problems in the graphics project. The period in which the system was quite unreliable was bothersome, as was the period of long evening downtimes, but such initial troubles are to be expected -- on the whole we have been very pleased with the performance of the system.

The sharing of community computer and clinical resources (via the SUMEX-AIM network) is essential to MISL's vitality. From the beginning, collaboration with the Rutgers Computers in Biomedicine group has been a central consideration in MISL planning. We have been very significantly influenced by Casimir Kulikowski's and Sholom Weiss's ideas concerning the causal state modeling of disease. The Rutgers group has provided us with detailed information concerning their work, largely via the SUMEX-AIM network. Recently, Weiss has made his diagnostic program for glaucoma available over the network. This will be used in a clinical setting by physicians at the Illinois Eye and Ear Infirmary.

## IV.A.2.e RUTGERS COMPUTERS IN BIOMEDICINE

Project: Rutgers Research Resource  
Computers in Biomedicine

Principal Investigator: Saul Amarel

(Grant NIH RR-00643-04, 3 years, \$285,240 this year)

## I. PROJECT GOALS AND APPROACHES

The fundamental objective of the Rutgers Resource is to develop a computer based framework for significant research in the biomedical sciences and for the application of research results to the solution of important problems in health care. The focal concept is to introduce advanced methods of computer science - particularly in artificial intelligence - into specific areas of biomedical inquiry. The computer is used as an integral part of the inquiry process, both for the development and organization of knowledge in a domain and for its utilization in problem solving and in processes of experimentation and theory formation.

The active Resource community consists at present of 26 members and 12 collaborators. Members are mainly located at Rutgers. Collaborators are located in several distant sites and they interact - via SUMEX-AIM - with Resource members on a variety of projects, ranging from system design/improvement to clinical data gathering and system testing. At present, collaborators are located at the Mt. Sinai School of Medicine, N.Y.; Washington University School of Medicine, St. Louis, Mo.; Johns Hopkins Medical Center, Baltimore, Md.; Illinois Eye and Ear Infirmary, Chicago, Ill.; College of Medicine and Dentistry of New Jersey (CMDNJ); and the University of Hawaii - Pacific Health Research Institute.

Research in the Rutgers Resource is oriented to "discipline-oriented" projects in medicine and psychology, and to "core" projects in computer science, that are closely coupled with the "discipline-oriented" studies. Work in the Resource is organized in four AREAS OF STUDY; in each area there are several projects. The areas of study and the senior investigators in each of them are:

- (1) Medical Modeling and Decision Making (C. Kulikowski, A. Safir).
- (2) Modeling Belief Systems (C. F. Schmidt).
- (3) Representations, Modeling and Hypothesis Formation in AI (S. Amarel).
- (4) Meta Description System (MDS) (C. V. Srinivasan).



In addition, the Rutgers Resource is sponsoring an Annual National AIM Workshop, whose main objective is to strengthen interactions between AIM activities, to disseminate research methodologies and results, and to stimulate collaborations and imaginative resource sharing within the framework of SUMEX-AIM. The Organizer of the first Annual Workshop (to be held at Rutgers on June 14 to 17) is C. Kulikowski; N. Sridharan is its Technical Director.

## II. AREAS OF STUDY; SUMMARY OF GOALS AND ACTIVITIES

### (1) Medical Modeling and Decision Making

Present projects include:

- (i) Development and clinical testing of the Glaucoma Consultation program based on a Causal-Associational (CASNET) model - as a collaborative project of the Ophthalmological network which we have initiated last year (Mt. Sinai Medical School, Washington Univ., Johns Hopkins Univ. and Illinois Eye and Ear Infirmary).
- (ii) Investigation of models of disease description based on a general semantic network representation, with associated strategies of diagnosis, prognosis and therapy. These models subsume a variety of representations and sub-models useful in general consultation in ophthalmological diseases and selected chronic diseases. A particular emphasis is placed on the analysis of the true course of disease and interrelationships between various subprocesses.
- (iii) In collaboration with The Mt. Sinai Health Care Computer Laboratory we are developing models of refraction strabismus and neuro-ophthalmology.
- (iv) In collaboration with the Pacific Health Research Institute in Hawaii we are developing models of chronic diseases: thyroid disease, hypertension and diabetes.
- (v) In collaboration with CMDNJ we are developing a model in hematology.

The following is a summary of accomplishments in this area:

- a. The Ophthalmological Network is functioning - consultation programs are now available through SUMEX-AIM to the four collaborating institutions.
- b. The Consultation System has been perfected by adding many details of diagnosis, pathophysiological states and new observations as the result of suggestions by the network participants.

- c. A file system for storing cases and providing a chronological model-based interpretation has been created.
- d. A set of programs to analyze the case histories is currently under development. When these are perfected they will be the initial step of a system for automatically incorporating experience into the consultation program.

This work is proceeding in close connection with an investigation in the AI area on models of grammatical inference and their relation to learning in causal structures of the type used in glaucoma.

The progress in the area of the Ophthalmological Network would be impossible without the facilities and support provided by SUMEX-AIM.

## (2) Modeling Belief Systems

The overall goal of this project is to develop a computer-based psychological model of how persons reason about the causes of human action. The common-sense notion of social causation which is used to understand intentional actions has served as the focus of this effort. Within this paradigm, the observer explains the intentional actions of others by attributing to the others the plans and motives which could have generated the observed actions.

To date we have:

- (a) defined the central concepts of such a notion of causality - that is, concepts such as person, act, plan, motive, belief, etc.
- (b) identified and studied various strategies for reasoning about observed actions;
- (c) investigated the implications of this type of reasoning for the organization of memory for such events;
- (d) and investigated questions of how, in the child, these concepts of plans and motives might develop.

Our goals for the immediate future are:

- (a) To continue our collaboration with the groups working in the AI and MDS areas in order to develop the model of social causation within the framework of MDS (the model is called BELIEVER);

- (b) to continue the work on the study of strategies for reasoning about action as well as to develop empirical procedures that will aid us in the definition and study of such strategies;
- (c) to continue our investigations on memory organization;
- (d) to continue the work on how such causal reasoning develops in the child;
- (e) to extend the theory of social causation to account for how persons evaluate, in a moral sense, the actions of others.

SUMEX-AIM is providing a common environment within which collaboration among the various investigators working on Belief Systems and those working on AI representation and MDS communicate and share programs. This has been particularly important during this year since the persons involved have been quite spread out geographically (Amherst, Mass., Cambridge, Mass., New Brunswick, N.J.).

### (3) Representations, Modeling and Hypothesis Formation in Artificial Intelligence

A major part of our effort in this area is oriented to collaborations with investigators in other Resource projects - involving applications of AI ideas and programs and also identification and initial exploration of new significant AI problems.

Present projects include:

- (i) Verifying consistency of a causal model of a physiological process (such as used in glaucoma) relative to an underlying mathematical model (N. Sridharan). Nlisp was the main tool used in this attempt. The current program provides a way of easily entering the mathematical model, performing checks for dimensional consistency of the variables entering into the equations, inferring dimensions of variables that were unspecified, using the graph depicting the relationships of variables and extracting the flow of determination in the variables. In order for the mathematical model to be consistent with a causal model, minimally one should ensure that the causal flow is compatible with the flow of determination.
- (ii) Interactive acquisition of domain Knowledge in modeling Belief Systems (N. Sridharan). A prompting program was written to accept descriptions of act schemata written in a natural english-like syntax. The prompting system is general and will accept any structural description for which it will provide prompting. One has also the capability to specify transformations to be made on different fields of the structure.

- (iii) Bringing up the FUZZY system of LeFaivre in INTERLISP and investigation of fuzzy problem solving approaches in medical decision making (R. LeFaivre). Implementation work is just starting here.
- (iv) Application of grammatical inference schemes to automatic adjustment of medical causal models on the basis of clinical data (A. Walker). A survey of grammatical inference techniques was made, from the applications point of view, and the most promising technique was related to our causal modeling approach in the medical decision area. This led to theoretical results in the area of stochastic grammars - an area which promises to provide formal support for our work in medical modeling.
- (v) Development of a grammatical inference system using a "developmental paradigm" (W. Fabens). This is a hypothesis formation system which attempts to change a given context free grammar so as to accommodate new sentences that cannot be derived from the given grammar. The system includes (a) a relaxation parser - which comes as close as it can to an interpretation of a given "deviant sentence", (b) a rule hypothesizer which uses such an interpretation to propose changes to the current grammar, (c) an intersection generator which tries to produce a sentence not legal before but legal after the rule modification process, and (d) a rule coalescer which summarizes with as little loss of gain in generality as possible the newly hypothesized grammar. We are currently concentrating on areas (c) and (d) and have developed programs for (a) and (b).
- (vi) Development and study of systems for theory formation in programming tasks (S. Amarel). Experimental systems are being developed for cases where the program to be formed is specified in terms of (a) a desired output condition, (b) program traces for specific input-output pairs, and (c) a set of input-output pairs. The current approach to (c) is to make the generation of candidate programs (hypotheses) responsive to a detailed analysis of shortcomings of previously generated candidates. In this project, major emphasis is given to problems of representation and to the effects of shifts between representations.

This project and the previous two projects are focusing on different AI approaches to hypothesis (theory) formation - an area which is essential to the automatic acquisition and improvement of a Knowledge base from experimental data.

SUMEX-AIM is providing the LISP environment needed for developing some of the AI programs in this area, and (more importantly) it is providing an effective communication environment for collaboration between AI people and investigators in the Medicine and Psychology areas.

#### (4) Meta Description System (MDS)

MDS is a tool for building knowledge-based systems. It has two modes of operation. One is the domain acquisition mode. In this mode the system acquires the KNOWLEDGE in a domain, in a given description schema. The other is the domain execution mode. In this mode the system uses the described domain Knowledge automatically for problem solving in the domain. Considerable progress was made during the last year in building the system facilities for domain acquisition. The MDS system is being implemented in INTERLISP on the SUMEX-AIM computer.

We are now using the MDS framework to develop the system design for BELIEVER (the Belief Systems Model of Social causation). We have also described the GLAUCOMA consultation system in MDS.

### III. AIM WORKSHOP

The theme of the first Annual Workshop is "Knowledge Based Systems in Medicine". The first day (June 14, 1975) will be devoted to a "General Session" which will provide an overview of current AIM activities and a broad forum for discussion. The following three days (June 15 to 17) will be devoted to discussions in depth of AIM designs, and to demonstrations of current systems.

The SUMEX-AIM system is essential for the Workshop. Most of the AIM programs will be running on SUMEX-AIM and accessed via TYMNET or ARPANET from Rutgers. The messages facilities of SUMEX-AIM have been most useful for planning, communicating and setting up the information pool for the AIM Workshop.

### IV. EXPERIENCE WITH THE SUMEX-AIM RESOURCE

In the last year, we have used the SUMEX-AIM resource for program development and testing and for communication between investigators.

INTERLISP, SITBOL, FORTRAN, Editing and Message Handling systems were extensively used. The message and linking facilities were used to provide a common environment for collaboration between several investigators in a project (some of whom were quite spread geographically). For example, in the BELIEVER project we have established a common MSG file that is shared for READ/APPEND access by workers in the project.

SUMEX-AIM has provided the linkages (communications, terminals) for the establishment of the network of collaborators in ophthalmology. The initiation of the net was delayed because of problems with getting prompt delivery of equipment from manufacturers.

The SUMEX-AIM Staff has been most helpful and cooperative. The system has been very useful to us in establishing collaborative research activities, in sharing information, and in providing a forum for linking and talking.

As remote users, we are especially sensitive to the communication facilities available to SUMEX-AIM. The in-wats line performed well. TYMNET has improved since its early days; however, its performance for users in the New York region (the Mt. Sinai group) remains relatively poor. The ARPANET connection is now providing a good communication medium. With the linking of both SUMEX-AIM and RUTGERS-10 on the ARPANET, we are now in a better position to work effectively on both systems (file transfers, communications). Plans are now underway to make the RUTGERS-10 more compatible with SUMEX-AIM, so that it can provide a more reliable and convenient network environment for our investigators and their collaborators. S. Levy is coordinating these plans.

A source of annoyance with SUMEX-AIM has been the frequency and duration of downtime. Improvements in this area are of special significance for our research collaborations - especially in interactions with our medical collaborators. From the point of view of program developers, the system appears now frequently overloaded. When the load average shoots to 10, response time is extremely poor and useful work becomes impossible. This is another area, where the availability of a compatible RUTGERS-10 system may help.

In conclusion, the SUMEX-AIM facility is now very much an integral part of our research environment. Several important components of our project are completely dependent on it. Taking into consideration that this has been a year in which the system was being brought up, and many changes have been taking place, the SUMEX-AIM resource has provided a very fine support for the Rutgers project.

#### IV.B INFORMAL PROJECTS

The following is a summary of the various "pilot" projects which have been admitted to SUMEX on a temporary basis pending development of a formal proposal. Many of these projects reflect initial efforts at formalizing analyses of experimental situations in preparation for the development of DENDRAL-like heuristic inference generation and modeling.

##### IV.B.1 STANFORD PILOT PROJECTS

##### IV.B.1.a ARTIFICIAL INTELLIGENCE APPLICATIONS IN GENETICS

Investigator: Prof. Cavalli-Sforza (Genetics)

(Grant NIH GM-20467-02, 3 years, \$49,092 this year)

The following are reports from members of Dr. Cavalli-Sforza's group who have used SUMEX this past year.

##### WAGENER

I have been using SUMEX primarily for REDUCE [A LISP system for manipulation of symbolic algebraic expressions developed by A. Hearn at Utah and transmitted to us via the ARPANET]. My research in theoretical population genetics involves formulation of sometimes quite complex mathematical models to describe various processes going on within and between human populations which may affect the genetic bases (genotypes) or expressions of these genotypes (phenotypes) in a population.

Part of my PhD thesis includes the analysis of the evolution of certain phenotypes that are influenced by interactions taking place between both genetic and environmental factors. The object of this research was to take a simple kind of interaction and show how tendencies for mating assortatively affect the evolution of these traits. When the mates are chosen nonrandomly, based on phenotypes, the family environments expressed in the population are affected. Then, when the distribution of expressed environments is changed, the distribution of phenotypes affected by these environments may also change. In the reduced form the recurrence equations of these models are quite complex, sometimes involving as many as 100 terms. I found little agreement between repeated attempts to verify the algebra by hand. By modifying the equations slightly I was able to verify conclusively these equations on REDUCE and was then able to program the model.

## AMMERMAN

We have been involved in the anthropological excavation of several sites in Calabria, Italy. The data will give information about the settlement patterns of Neolithic man as part of our analysis of biological-anthropological and cultural adaptations. Correlation of sites involves the use of several types of data: coordinates, material found, period of material, elevation, land form, source of nearest water, soil, geology classification and present land use. The second phase of data collection is in progress as a preliminary to more comprehensive modeling efforts.

## THOMPSON

I have attempted to make a comprehensive simulation to generate data on the evolution of chromosomes in a finite population. The simulations are stochastic and involve mutation, many types of selection, epistasis, and recombination. The affects of finiteness on the evolution of certain traits (drift) may be studied. Also the distribution of ages of mutants and selection on new mutants is generated. The model is written in FORTRAN.

Investigator: Prof. J. Lederberg

Other Genetics Dept. Projects: Molecular Genetics  
& DNA Segments

My own laboratory group has been using SUMEX to help start the development of new A-I programs for hypothesis-formation, automated explanation and assistance in the induction and planning of new experimental procedures in our laboratory work in molecular genetics. We have therefore been using SUMEX wherever it was feasible for supporting our day-to-day laboratory work AND where this also contributed to establishing the requisite knowledge base.

The modelling of DNA subjected to segmentation by restriction endonucleases at randomly occurring specific sites of bacterial DNA has already been invaluable in the understanding of our experimental findings. Cf:

Harris-Warwick, R., Ehrlich, S., Elkana, Y., & Lederberg, J., "Fraction and Purification of Bacterial Genes by Segmentation of DNA with EcoR1 Endonuclease and Agarose-gel-electrophoresis", Proc. Nat. Acad. Sci., U.S., In Press [June '75].



#### IV.B.1.b INFORMATION PROCESSING PSYCHOLOGY PROJECT

Principal Investigators: Prof. E. Feigenbaum (Computer Science)  
and Prof. H. Cohen (U. C. San Diego)

(Grant application in preparation)

##### I. Abstract of Project Goals and Activities

The general goal of this research is the development of information processing models of human problem solving, learning, and memory, using techniques of computer simulation. In the spirit of much previous work of this type by the artificial intelligence research community (some of which has been sponsored by NIH and NIMH), this work is to be thought of as application of AI concepts and techniques to theory construction in Psychology. The effort currently most active within the IPP Project is a pilot study of perceptual, memory, and performance processes involved in certain types of construction of visual forms (involving composition and production of freehand line drawings). It is being carried out in collaboration with Prof. H. Cohen, Dept. of Visual Arts, Univ. of Calif., San Diego.

##### II. Summary of Project Accomplishments

The primary activity of the IPP project during the year has been the pilot study mentioned in I above. The "pilot" intention is to prepare the work to be discussed below for formal grant proposal submission to NIMH. If that project is eventually approved, it will apply for separate status at SUMEX.

The current study of art-making behavior is concerned with the modeling of how significance (= "meaning") is ascribed to symbols and groups of symbols, and how symbols are manipulated for the generation of significance.

Previously, computer simulation models have been used to study a subset of meaning which belongs in the domain of communication and concerns itself with verifiable fact. Thus, problem-solving models are expected to find answers which are demonstrably "right", inference models to draw inferences which are verifiable, conversation models to "understand" what the conversant intended to communicate. The reasons for this bias towards "well-defined" problems are clear and reasonable.

The more general view adopted in this work involves the propensity in human cognition to GENERATE meaning. Under study is the view that "creative" modes of intellectual behavior may be thought of as game-playing in the domain of meaning-generation.

The project work breaks down into the following stages:

1. The delineation of a group of drawing protocols, the exercise of which will trigger the meaning-generating propensity of the viewer; that is, which will present the viewer with an invitation to play with the significance of the drawing. In order to demonstrate the existence of this function in the viewer, and its importance in the whole "meaning" transaction, it is required that the model exhibit no intentionality with respect to meaning.
2. Once the separation of (viewer) generated meaning from (artist) intended meaning is established, it will be necessary to take into account that artists would normally exhibit intentionality - whether or not that intentionality is the source of "meaning" for the viewer - and that this intentionality must be expressed in relation both to the artwork and to the artist's view of the world. We will then need to consider in what way the artist's view of the world might function to provide determinants to his art-making behavior. This last question is particularly important, since art-making performance does not appear to be as goal-oriented as performance in chess, infectious disease diagnosis, or most other tasks studied by AI researchers.

The project is currently (predominantly) in Stage 1, though an effort has been made to develop a formalism which will support projected work in Stage 2 without monumental reworking. In order to accomplish this some preliminary work has been done on a simple associative memory model of a conventional semantic net form, and a program has been written which will permit the user to build up a large "sample" memory for testing purposes very rapidly.

The balance of the effort of the last few months has been directed to the clear definition of what a drawing protocol might be. Since this proceeded from the constraint of non-intentionality with respect to significance, the program has never at any stage adopted a lexical approach. In place of morphological units like lines and squares and circles the program performs with perceptual units like figure/ground differentiation and inside/outside differentiation, and with task units like pathfinding. Closed forms of considerable complexity are generated from the notion of closure, not from a lexicon of polygons. CLOSURE is thus an example of a protocol.

This protocol-oriented formalism has been generalized into a production system in which the left sides list states of protocol history and the right sides give appropriately weighted tables of admissible protocols. The production system thus embodies the model's explicit knowledge of image-making, and is in a form which should eventually permit extension to world knowledge.

The list of available protocols is now being developed vertically to include a number of REPETITION protocols, thus opening up the possibility of hierarchical performance; e.g. the application of the CLOSURE protocol repetitively to a recent protocol sub-history. (a ring of rings, etc.)

The replacement of the traditional lexicon-based morphology by a perception-informed drawing-protocol-based morphology promises to prove a powerful tool in the reexamination of a wide range of primitive image-making. A preliminary attempt is now being made to apply it - by hand - to a formal description of the Chalfont Valley (Calif.) group of Indian petroglyphs.

### III. Comments; Assessments; Correlation with Expectations

The interaction with the SUMEX machine and the SUMEX organization has been eminently satisfactory; no change is expected or desired.

As is the case for most users, the need for more file storage is critical, but we have no special demands in this dimension.

We do not envision use of the system in excess of the usage previously discussed; plan no new departures that will involve significant computer use; and will remain in pilot project mode until submission and approval of our grant request later this year.

## IV.B.1.c      AIM RESEARCH - UNIVERSITY OF ROCHESTER

Investigators: Drs. Feldman, Rovner, and Low  
Rochester University

(Grant NSF DCR74-24203, 2 years, \$149,956 total and  
Sloan Fdn. 74-12-5, 3 years, \$120,000 this year)

The Rochester group has a number of projects in connection with the SUMEX-AIM facility. These range from the use by medical students of existing systems to research on programming languages and techniques for Artificial Intelligence. The group itself is quite new and will be initiating new research programs over the next year as the staff builds and our research plans develop sharper focus. This note will describe existing programs and then briefly discuss anticipated efforts. In addition to our research work on SUMEX, we have worked on system development primarily in BCPL and SAIL.

## A. Evaluation of Existing Systems

Some second year medical students have been evaluating MYCIN from the point of view of a physician under the guidance of Dr. Roy Steigbigel, a specialist in infectious disease and Dr. Charles Odoroff, Chairman of the Biostatistics program.

The students have tried a number of cases based on their course work and on hospital experience suggested by Dr. Steigbigel. There has been some communication with Bruce Buchanan about the system problems, so the beginnings of an interaction are there. An assessment of the applicability of MYCIN to clinical problems is due in early June. A copy will be forwarded when it is available.

## B. Automatic Choice of Data Representation

Abstract data structures such as sets, lists and relations are being used more and more within programming languages for Artificial Intelligence. They ease programming by providing good models with which the programmer can express his problem. However there is no one fixed representation of any such abstract data structure which is optimal for all programs. Each set representation (such as linked lists, binary trees, boolean arrays and so forth) has its own properties. Depending on how a program uses an abstract structure, one representation may be far superior to others in terms of storage occupied by the data structure and/or the time needed for manipulating it. For example, if the operations on a particular set are just membership testing, insertion and removal of elements then a boolean array may be the best representation. However if the size of the set is normally small but the number of potential elements is large, this representation is very expensive in storage requirements. If the set

is iterated through (FOREACH element in the set do some operation), the time needed for iterating through the set is proportional to the number of potential elements not the actual number of elements. Other representations have different storage and time requirements. The problem is to choose an appropriate representation for each set and list of SAIL program instead of having one fixed representation which is always used. A system for doing this was developed by Jim Low, now at the University of Rochester, as part of his dissertation research at Stanford. This system has, in the last few months, been adapted to run under TENEX at SUMEX. This system uses information obtained by static analysis, of the SAIL program, dynamic statement execution counts, and information supplied by interrogation of the user.

We hope to continue to enhance this system by improving the decision making processes and extending the capabilities to allow change of representation.

#### I. More Efficient Implementation of LEAP

The associative aspect of LEAP of the programming language SAIL is a data base of 3-tuples. LEAP allows the programmer to query the data base in any of seven ways characterized by the those fields left unspecified for the search. In the following A, O, V represent specified fields of the 3-tuple.

- (1) (A,O,V) (does a totally specified 3-tuple exist)
- (2) (?,O,V) (find all 3-tuples with specific 2nd and 3rd components)
- (3) (A,?,V)
- (4) (A,O,?)
- (5) (?,?,V)
- (6) (A,?,?)
- (7) (?,O,?)

The current implementation of LEAP uses hash-coding techniques to perform searches (1) and (4). An inverted file scheme (on the third component) is used to perform searches (2), (3) and (5). The data structure used in representing 3-tuples does not allow direct searches of types (6) and (7). The current implementation simply iterates through all possibilities of the first position for search (7) (all possibilities of the second position for search (6)) and then performs a search of type (4). Thus, with the current implementation, searches of types (6) and (7) are orders of magnitude slower than the other searches.

Many techniques are known for speeding up these searches, but most either use more space (by redundantly storing information), or slow down the other searches significantly. Richard Rashid, a graduate student successfully developed a scheme which avoids these pitfalls. He has accomplished this by changing the hash coding algorithm so that the hash code (from hashing the attribute and object together) contains some information about its parents. He thus can perform searches of types (6) and (7) by iterating through the parts of the hash code normally supplied by the first component in search (7) or the second component in search (6). The size of this set of possibilities is on the order of  $\sqrt{N}$  where  $N$  is the number of buckets as opposed to  $N$  in the current LEAP implementation.

We were concerned that this new hashing algorithm might not be as good as the old for searches of types (1) - (5). In preliminary test runs with complex programs involving a large use of LEAP, however, we have found more than a 10% improvement -- far more than we would have expected from the improvements in searches (6) and (7) alone. We have therefore hypothesized that the new hashing algorithm is actually better for the normal uses of LEAP than the old and hope to begin testing that theory in the near future. If our preliminary results are shown to be valid, we expect to incorporate this new algorithm into SAIL itself sometime in the near future.

### C. Automatic Choice of Associative Data Structures

Many computer applications, especially in AI and information retrieval, deal with relational data and use associative retrieval techniques. The choice of a good associative data structure for a given program is often crucial; a poor choice could well lead to gross inefficiencies in storage space and search times. Furthermore, there is no "general purpose" associative data structure that does the job. Although many general schemes work, virtually every program that would use one has some particular behaviour pattern for which the general data structure is sub-optimal. Significant improvements in performance are usual after changing from a general scheme to one that is chosen to match the requirements of the program at hand.

We are using SUMEX to study ways to systematize the selection of such data structures. We are developing techniques to model the behaviour of programs, the structure of data bases, and the properties of an important class of representation techniques. The goal is to learn how to build a "smart compiler" that will automatically select an associative data structure for a given program. Such a system will analyze a given program, ask questions of the program's designer, analyze examples of the execution of the program, and then compose a data structure package from a library of data structure techniques. The selection will be based on the cost (for a given program) of candidate data structures, given by a function of their expected storage space and execution time requirements.

We started using SUMEX in a serious way in December, 1974. Since then, we have

1. implemented a library of associative data structure techniques (for n-ary relations), and an interactive program for composing a data structure package from this library. The techniques include hash tables, property lists, records, partially and fully inverted files, and methods of sharing storage between hash tables and inverted files. The programs are written in BCPL.
2. modified Jim Low's system for analyzing a SAIL program to include:
  - a. extensions to the SAIL syntax to allow n-ary relations and n-ary associative retrieval
  - b. various extensions to the modules that model the ways in which the given program uses its relational data.

We are currently studying the properties of the various representation techniques in our library, and the model of program behaviour that is derived by the analysis programs. We expect to begin soon to formulate and implement heuristics for suggesting representation techniques from usage patterns that appear in the model.

#### Comments on SUMEX:

My overall impression of the combination of TYMNET and SUMEX is that "it is OK: I can get my work done". In response to your request, I will write down my most strongly held criticisms, though I really don't feel them very strongly, since I am able to work this way.

1. TYMNET is down too much of the time, and crashes too often.
2. SUMEX is down too much of the time, especially during the day on weekends.
3. TYMNET echoing is a mess. TYMNET echoing behaviour should be adjustable under program control. I would often rather wait for echoing (and type ahead) than be confused by messed-up prompting and echoing, for example.
4. Response to control C is painfully slow.

On the other hand,

1. the load average is never too high for my work,
2. the people with whom I have dealt have been helpful and courteous

#### D. Planning and Acting Under Uncertainty

Any attempt to apply Artificial Intelligence methods to medicine will have to deal with the uncertainties, risks and costs involved in a clinical situation. Traditional AI techniques have been developed for purely symbolic situations where these issues were not so important. Traditional automated diagnosis techniques have involved the use of decision theory to solve these problems but have foundered on the huge trees required for realistic problems. We are engaged in fundamental and applied research on methods for combining heuristic and decision-theoretic methods.

#### E. Plan for the Immediate Future

When the University of Rochester set up the Computer Science Department in 1974, interdisciplinary studies were one of the paramount goals. The Medical School is one of the strongest parts of the University and already has strong ties with the Computer Engineering program. These factors plus the interests of the Computer Science faculty make a concerted effort on AI and Medicine very attractive. It is likely that the academic year 1975-76 will see joint appointments with Radiology and with Obstetrics. This will enable us to greatly expand a small current program in intelligent processing of medical images. We also intend to apply the work on planning and acting (section D) in an appropriate clinical context. The ideal would be to find an area where our image processing and problem solving efforts would be symbiotic. When the staff gathers next fall, we will make a concerted effort to define a research plan in the AIM domain.



## IV.B.1.d NATURAL LANGUAGE UNDERSTANDING

Investigator: Prof. R. Lindsay  
University of Michigan

(Financial support from University of Michigan)

This SUMEX account has been active since January 1975. The user is Kathie Gourlay in Ann Arbor, Michigan. Mrs. Gourlay is an assistant to Professor Robert K. Lindsay, visiting Stanford this year.

In the three months of this project's existence, our main objective has been to familiarize Mrs. Gourlay with SUMEX. She has been learning TENEX, SOS, and INTERLISP. Access is over TYMNET's Detroit node. We have found SNDMSG and LINKing to be very useful devices.

No substantive research has yet resulted. It is hoped that we will continue next year, as a non-Stanford SUMEX project. That work, for which use to date has been groundwork, will be concerned with natural language processing, particularly the development of memory structures for word meanings, and (perhaps) languages for organizing large data bases. In addition, Professor Lindsay will continue to work, via SUMEX, with Dr. Engelmores and Dr. Freer (UCSD) on the protein crystallography project.

We have concluded that some means of obtaining remote listings at reasonable speeds is essential. We are attempting to use a Centronix model 308 teletype for this purpose, but have not yet worked out the details. If successful, other installations might be interested in our experience with this device, which should be capable of listing over phone lines (not TYMNET) at 120 cps; the price of the model 308 is between \$3000 and \$4000, depending on options. A high speed input device such as a cassette based terminal would also be useful as a means of reducing phone charges. We have not looked into this, but mention it in case others have a similar need. The communications features of the system have been very useful, as noted above.

## IV.B.1.e QUANTUM CHEM. INVEST. OF HEME PROTEINS AND FERREDOXINS

Investigator: Dr. Gilda Loew (Genetics)

(Grant NSF GB-40105, 2 years, \$18,000 this year)

I. Current projects and goals involving use of the PDP/KI-10 computer of SUMEX are:

- A. Study of antiferromagnetic behavior of the 2-iron family of sulfur containing proteins such as ferredoxin using
  - 1. The method of electric field gradient
  - 2. The method of electron correlation
  - 3. And the method of magnetic susceptibility;
- B. Calculations of quantum electromagnetic properties including
  - 1. Nuclear-electron spin-spin coupling
    - a. Fermi contact term which includes ligand contribution
    - b. Dipolar tensor which excludes ligand contribution
  - 2. Electric field gradient tensor and quadrupole splitting
  - 3. Zero-magnetic-field splitting as a result of spin-orbit coupling, and  $g$ (gyromagnetic ratio) values for the active site of such iron-containing compounds as hemoglobin, ferredoxins, ferrichrome-A's, mycobactins and of ferrecines.
- C. The goal for next year will be to complete the studies described above.

II. Summary of project accomplishment by means of SUMEX

- A. Our investigation of ferredoxin using the method of EFG (I.A.1) strongly suggests the existence of antiferromagnetic coupling; hence, we shall pursue the subject further by means of I.A.2 & 3;
- B. We have successfully accounted for the set of properties described in I.B for both of the oxidized and reduced state of the ferredoxin compounds at their active sites
- C. We have obtained a picture of antiferromagnetic coupling between the oxygen substituent and the iron at the hemoglobin active site from the results of EFG calculations consistent with that previously postulated from electronic spectra;

- D. We have greatly extended the flexibility of the 1-electron g-value program to calculate related properties for many-electron system using the hole-excitation concept;
- E. Regarding the study of the ferrocenes, a sandwich charge-transfer complex with iron in the middle, we have shown that semi-empirical methods give parallel results to ab-initio methods in describing the ionization process of the compounds and our calculated EFG show the collapse of the quadrupole splitting upon ionization in agreement with experiment;
- F. Regarding the study of the ferrochrome-A, a type of iron transport compounds, our preliminary results on g-values and spin-spin coupling indicated that the latter can be vastly improved by using spin-mixing wavefunction from the former as basis function;

### III. Comments

- A. The interactive nature of the SUMEX computer has been a great help in the course of our effort to probe the validity of our methods of calculations. It has also helped a great deal in the use of the 1-electron and 5-electron g-value programs where extensive parameterization is required and immediate feedback is extremely valuable owing to the uncertainty of the values of the parameters;
  - B. DDT has been a very convenient debugging tool for us, and so are the TENEX file-handling facilities;
  - C. We hope to take advantage of SAIL's excellent capabilities for handling utilities to construct
    - 1. An information retrieval system to keep track of and quick reference of our increasing amount of results;
    - 2. Tape utilities programs for storage and transporting;
  - D. We suspect that there is a bug in one of the FORTRAN library routines such as CABS and there seems to be questions about the PA10/50's interpretation of the UUO's;
- IV. We have noticed that our needs to use the SUMEX computer have sometimes exceeded our original expectations and hence our original agreement[\*] with SUMEX due to pressure from our research schedule. We have also tried whenever possible to use the computer at nights and abide with our agreement. In the future, we shall also try to use batch mode whenever possible. Finally, we are very grateful for our privilege to use the SUMEX computer and other facilities

[\*] These studies are part of a trial collaboration to

investigate ways of introducing heuristic approaches for more efficient structure determination. In view of the large potential of other aspects of Dr. Loew's work for large "number-crunching" consumption of CPU resources, Dr. Loew has agreed to conduct here work on a non-interference basis. The great bulk of her computation is done on other machines.

## IV.B.1.f AUTOMATIC INTERMACHINE PROGRAM TRANSLATION

## PILOT PROJECT SUMMARY FOR SUMEX ANNUAL REPORT

Investigator: Mr. J. Warren (Elect. Engrng.)

(Research Assist under grant NSF GJ-41644, 2 years,  
\$43,748 total, Prof. T. Bredt, Stanford EE, P.I.)

## 1. UPDATED ABSTRACT:

This work concerns the development of a methodology and set of tools to assist in automatic transporting of computer programs between computers in any given class of machines, initially a large variety of minicomputers. The approach being used is that of "automatic programming":

A translating system is under development that will accept a description of the instruction set of any machine in the target class. From that description, it will derive or "learn" instruction sequences that perform more complex functions. It will then apply what it has learned to the translation of high-level language programs, through an intermediate form that is phrased in terms of those "more complex functions", into object code for the desired target machine. The translator creates the translation algorithms, on its own, utilizing only the static instruction set description, and built-in knowledge of the semantics of the primitives in which both input and instruction sets are described.

Essentially, this project involves the application of the concepts and techniques of artificial intelligence to a very low-level intellectual activity, and is in hopes of obtaining results that will be of significant practical value, both within and beyond the area of biomedical computer applications.

## 2. SUMMARY OF ACCOMPLISHMENTS OVER THE PAST YEAR

(Use of SUMEX facilities only began in earnest around the middle of February.) This translation system is still in a relatively early stage of design. Only recently have parts of the design been sufficiently solidified to allow some entry of data into SUMEX facilities, and the initiation of construction of some of the programs that will ultimately assist the proposed translation system.

## 3. COMMENTS CONCERNING SUMEX FACILITIES

(Use of SUMEX facilities only began in earnest around the middle of February. It took several weeks of part-time activity to learn TECO sufficiently well to use it as a production editor. Current efforts are to learn SAIL sufficiently well to use it as a production compiler.)

To date, the majority of this investigator's time spent on SUMEX has been expended in familiarization with the system, and choosing system facilities to be used (e.g. choosing an editor, and choosing a high-level language in which to construct the translator system). The SUMEX facility is a delight to use; it is much preferable to any of the several other interactive systems with which this researcher has had experience, both in general and for this particular project. Further, the staff have been consistently cooperative and helpful concerning system facilities and problems. The only problems noted have concerned telephone communications and have been telephone company hardware problems; not SUMEX problems. Documentation of some software has been somewhat cumbersome (e.g. SAIL documentation [\*]), however, this is a problem that is virtually universal among general-purpose computing facilities and is by no means unique to SUMEX. Further, it is fairly evident that significant efforts are being made to upgrade the weaker documentation.

[\*] "This is universally agreed. We are currently working on a TENEX-oriented revision of the NIH-DCRT "Beginner's SAIL Manual".

#### IV.B.2 NATIONAL PILOT PROJECTS

There are no pilot projects charged to the allocation of the AIM Executive Committee resource at this time. The management committees are considering a number of projects which may be enabled in this category in the future.

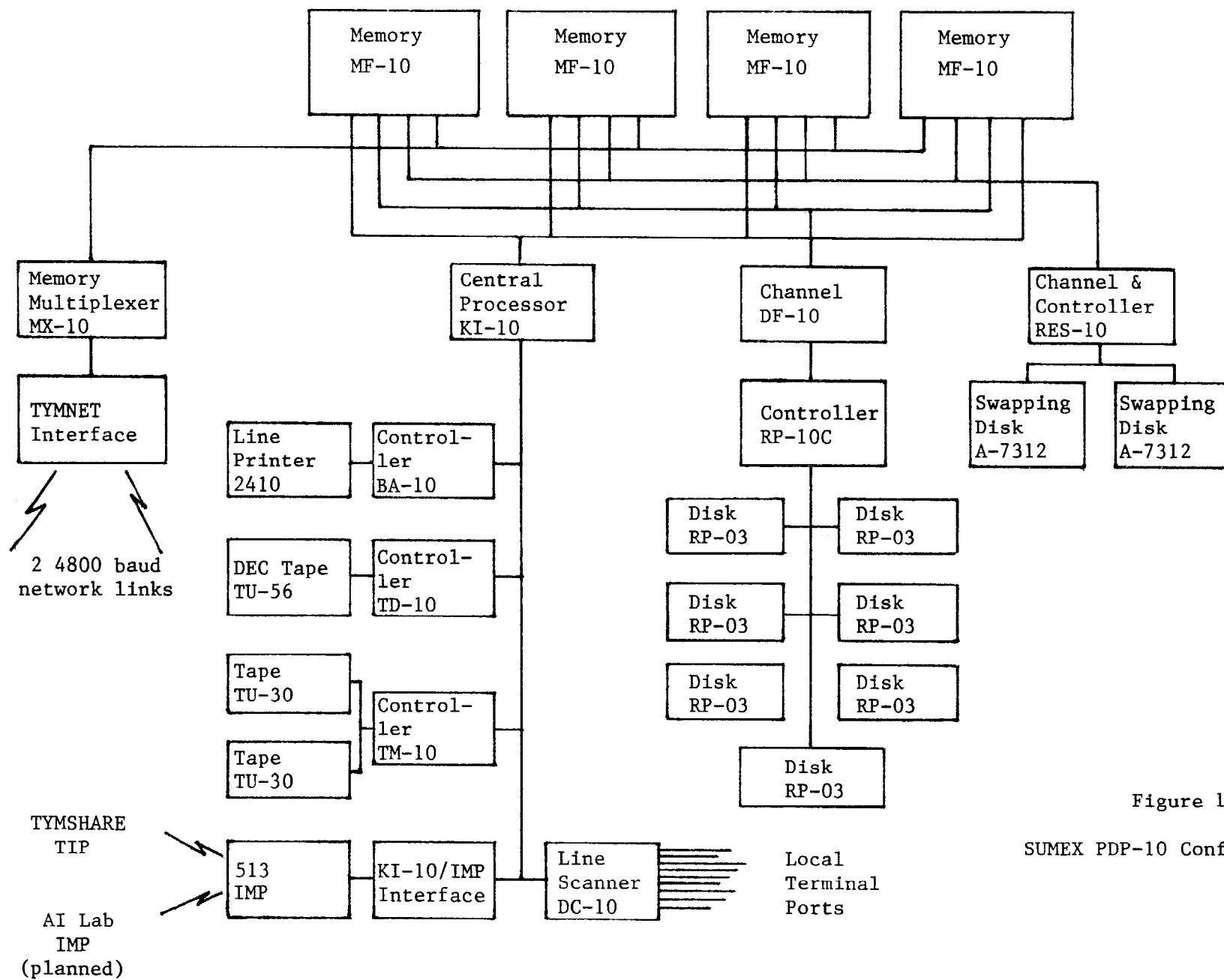


Figure 1.  
SUMEX PDP-10 Configuration

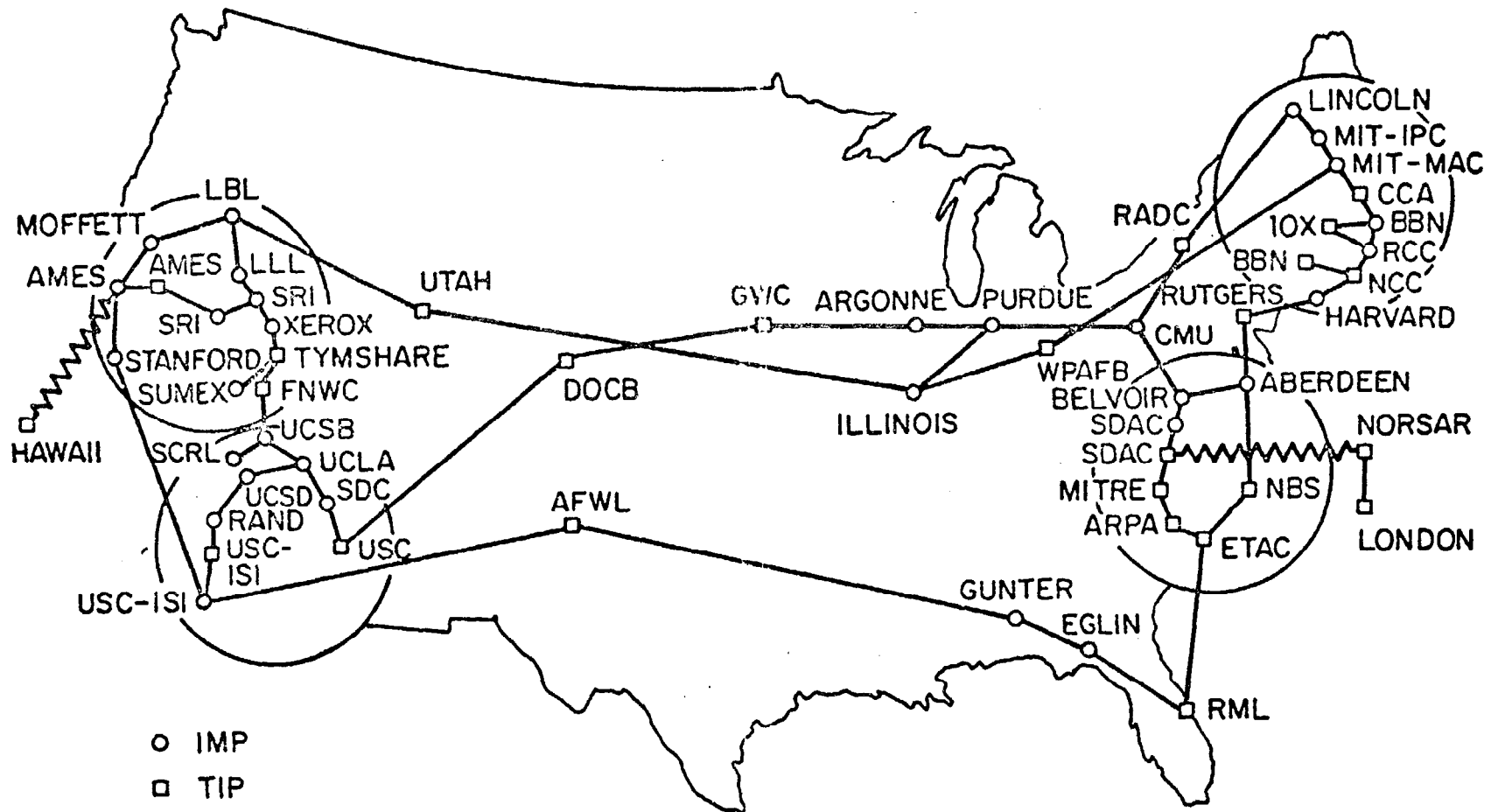




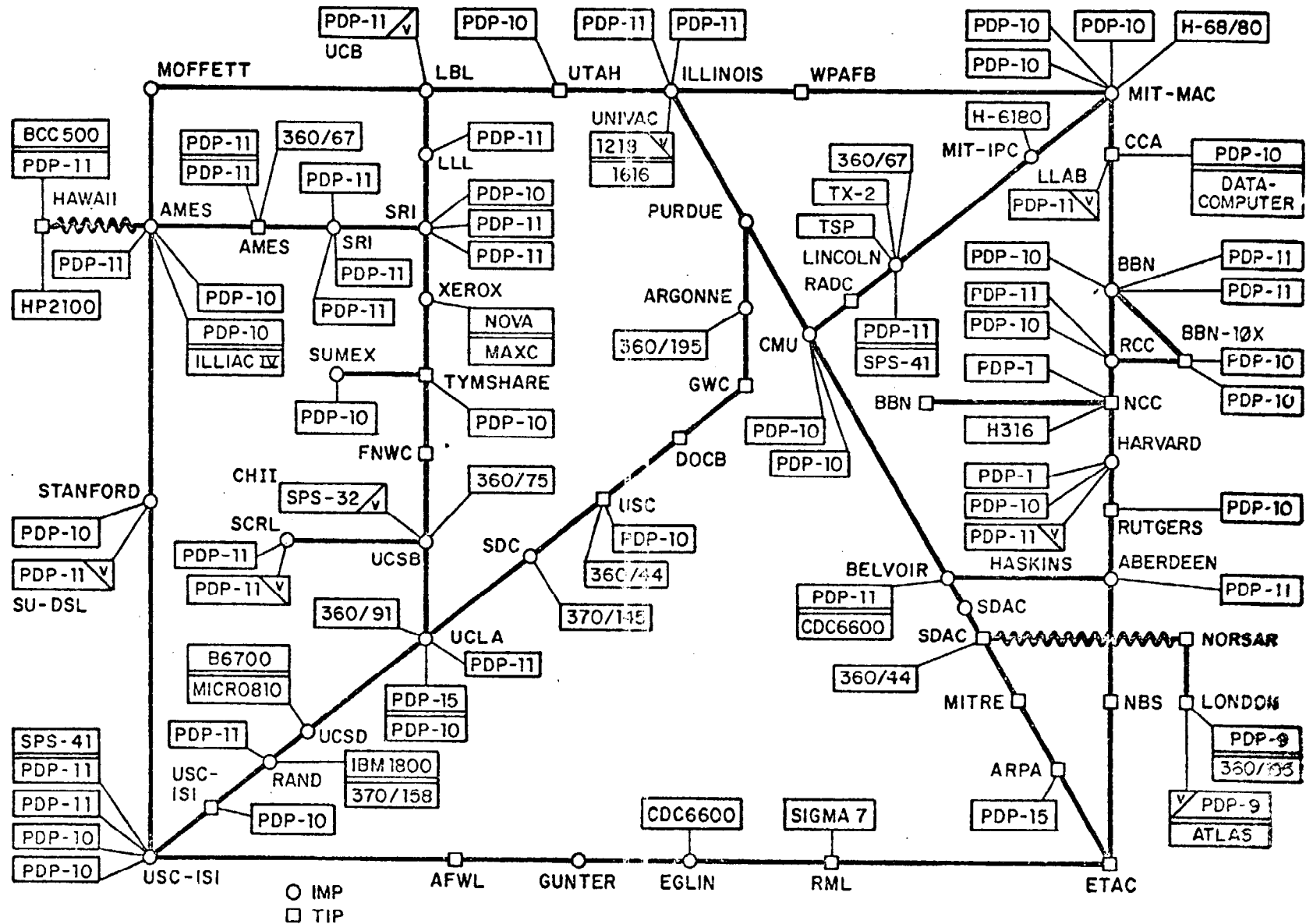
Figure 3.

## ARPA NETWORK, GEOGRAPHIC MAP

APRIL 1975



ARPA NETWORK, LOGICAL MAP, APRIL 1975



APPENDIX A

AI Overview by E. A. Feigenbaum

ARTIFICIAL INTELLIGENCE RESEARCH

What is it? What has it achieved? Where is it going?

Excerpt from a report by  
Professor Edward A. Feigenbaum  
Stanford University

## INTRODUCTION

In this briefing, these questions will be discussed as succinctly as possible:

- I. What is the scientific field of artificial intelligence research, as seen from various viewpoints? What are the general goals of the field?
- II. What are its practical working goals? What are some achievements relative to these goals (circa 1973)?
- III. What steps (new goals, problems, potential achievements) seem to lie ahead, within a five year horizon?

## ARTIFICIAL INTELLIGENCE (alias INTELLIGENT COMPUTER SYSTEMS):

### General View;

Artificial Intelligence research is that part of Computer Science that is concerned with the symbol-manipulation processes that produce intelligent action. By "intelligent action" is meant an act or decision that is goal-oriented, arrived at by an understandable chain of symbolic analysis and reasoning steps, and is one in which knowledge of the world informs and guides the reasoning.

Some scientists view the performance of complex symbolic reasoning acts by computer programs as the sine qua non for artificial intelligence programs, but this is necessarily a limited view.

Yet another view unifies AI research with the rest of Computer Science. It is an oversimplified view, but worthy of consideration. The potential uses of computers by people to accomplish tasks can be "one-dimensionalized" into a spectrum representing the nature of instruction that must be given the computer to do its job. Call it the WHAT-TO-HOW spectrum. At one extreme of the spectrum, the user supplies his intelligence to instruct the machine with precision exactly HOW to do his job, step-by-step. Progress in Computer Science can be seen as steps away from that extreme "HOW" point on the spectrum: the familiar panoply of assembly languages, subroutine libraries, compilers, extensible languages, etc. At the other extreme of the spectrum is the user with his real problem (WHAT he wishes the computer, as his instrument, to do for him). He aspires to communicate WHAT he wants done in a language that is comfortable to him (perhaps English); via communication modes that are convenient for him (including perhaps, speech or pictures); with some generality, some abstractness, perhaps some vagueness, imprecision, even error; without having to lay out in detail all necessary subgoals for adequate performance - with reasonable assurance that he is addressing an intelligent agent that is using knowledge of his world to understand his intent, to fill in his vagueness, to make specific his

abstractions, to correct his errors, to discover appropriate subgoals, and ultimately to translate WHAT he really wants done into processing steps that define HOW it shall be done by a real computer. The research activity aimed at creating computer programs that act as "intelligent agents" near the WHAT end of the WHAT-TO-HOW spectrum can be viewed as the long-range goal of AI research. Historically, AI research has always been the primary vehicle for progress toward this end, though science as a whole is largely unaware of the role, the goals, and the progress.

## HISTORICAL TRACE

The Working Goals of the Science;  
Progress toward those goals;

The root concepts of AI as a science are 1) the conception of the digital computer as a symbol-processing device (rather than as merely a number calculator); 2) the conception that all intelligent activity can be precisely described as symbol-manipulation. (The latter is the fundamental working hypothesis of the AI field, but is controversial outside of the field.) The first inference to be drawn therefrom is that the symbol-manipulations which constitute intelligent activity can be modeled in the medium of the symbol-processing capabilities of the digital computer.

This intellectual advance--which gives realization in a physical system, the digital computer, to the complex symbolic processes of intelligent action and decision--with detailed case studies of how the realization can be accomplished, and with bodies of methods and techniques for creating new demonstrations--ranks as one of the great intellectual achievements of Science, allowing us finally to understand how a physical system can also embody mind. The fact that large segments of the intellectual community do not yet understand that this advance has been made does not change its truth or its fundamental nature.

Three global "working goals" have dominated the AI field for the 17 years of its existence. These are:

1. Understanding heuristic search as a processing scheme sufficient to account for much intelligent problem solving behavior; and exploring the scope and pervasiveness of heuristic problem solving.
2. Semantic information processing: developing precise formulations of "understanding" by programs, and "meaning" of symbols that are input or stored; the acquisition, storage, and deployment of knowledge of the world in the service of symbolic problem solving.

3. Information Processing Psychology: developing precise models of human behavior in symbolic-processing tasks.

The first two goals represent the fundamental paradigms that have dominated the field. The third cuts across these orthogonally, and involves intense interdisciplinary contact with Psychology, and Linguistics.

#### GOAL 1. HEURISTIC SEARCH, HEURISTIC PROGRAMMING, SYMBOLIC PROBLEM SOLVING PROGRAMS

In the first decade, the dominant paradigm of AI research was heuristic search. In this paradigm, problem solving is conceived as follows: A tree of "tries" (aliases: subproblems, reductions, candidates, solution attempts, alternatives-and-consequences, etc.) is sprouted (or sproutable) by a generator. Solutions (variously defined) exist at particular (unknown) depths along particular (unknown) paths. To find one is a "problem". For any task regarded as nontrivial, the search space is very large. Rules and procedures called heuristics are applied to direct search, to limit search, to constrain the sprouting of the tree, etc. While some of this tree-searching machinery is entirely task-specific, other parts can be made quite general over the domain of designs employing the heuristic search paradigm. Two notions are critical. The first is that problem solvers generally face a "maze" of alternative courses of decision and action that is huge compared with their processing resources. The second is the use of heuristic knowledge to steer carefully through large mazes toward a solution seeking the plausible and potentially fruitful avenues, avoiding the absurdities and the high-risk paths. Heuristic knowledge is usually informal knowledge--to be distinguished from formal knowledge that is assertable with the rigor of proof. Polya, the famous mathematician who wrote *Patterns of Plausible Inference* and other books on problem solving, calls heuristic reasoning "the art of good guessing." Heuristic knowledge is often "common sense" knowledge of the world, rules-of-thumb for generally acceptable performance, or rules of good practice in specific situations. When we speak of the "expertise" of an expert, and the "good judgment" he brings to bear on complex problems in his domain, we often are speaking of the heuristics he has developed to search effectively.

Provocative essays by Polya notwithstanding, the first serious and detailed studies of heuristic problem solving ever done by Science were done as AI research in its first decade. As with any other science, progress came by the detailed examination of specific cases, from which gradually emerged both a broad picture of the nature of the phenomena being studied and, within this, more formal theories for specific parts.

Three sub-goals of heuristic programming are discernable.

SUBGOAL 1A. Demonstrate sufficiency of heuristic search for tasks of intellectual difficulty.

These heuristic programming efforts dealt with almost "pure" symbolic reasoning tasks (i.e., tasks not requiring much coupling to real-world knowledge), and used inference schemes that were either ad-hoc or of limited scope. Notable successes during this "prove-the concept" phase were: the Logic Theory Program, that proved theorems in Whitehead & Russell's propositional calculus; the Geometry Theorem Proving program, that proved theorems in Euclidean geometry at a level of competence exceeding that of the excellent high school geometry student; the Symbolic Integration program, that solved college freshman symbolic integration problems about as well as MIT freshmen; chess-playing programs that play respectable "club player" C or B Class chess; a checker playing program that was virtually unbeatable, except by the country's top few players (notable also for remarkable self-improvement in performance by analysis of its own play and "book-move" good play); and a number of competent management science applications (assembly-line balancing, warehouse location, job-shop scheduling, etc.).

To recapitulate briefly: the key concepts are: search in problem solving; and the use of generally informal knowledge to guide search effectively. The AI community was the first to devote serious scientific effort to developing the idea of the use of informal knowledge in problem solving, with notable successes. Few in Science recognize that this achievement has been made and is ready for exploitation.

SUBGOAL 1B. Generality in Problem Solving Programs

Generality here means the use of a small set of problem solving methods of wide applicability to solve problems of many different types. Each of the problems posed is stated to the program in a particular representation (or framework) with which the set of methods is constructed to handle.

The subgoal of generality arises first as a reaction to the array of "specialty" programs mentioned above; second, from the general observation that the ability to do a wide range of tasks is a special touchstone of intelligence; third, from a direct assessment that as the diversity and heterogeneity of the tasks handled by an agent increases, the likelihood that it can do them all without intelligent action decreases; and fourth, from the argument that any ultimate intelligent agent must have wide generality, since it must take the world and its problems as they come without any intermediary, making generality an important independent desideratum.

This subgoal was pursued with vigor for ten years in a number of projects, was important for its feedback value in clarifying issues for the AI field, and has temporarily (at least) been put back on the shelf as the field begins to explore knowledge-based problem solvers and issues in the representation of knowledge.



There were two discernable subthemes. The first was an attempt to create abstract heuristic search methods that were divorced from any particular content. Examples were: the General Problem Solver, which used a variant of heuristic search known as means-ends analysis; MULTIPLE, which introduced adaptivity in the selection of what subproblem to choose "next" in a search; and REF-ARF, which extended the generality of ordinary procedural programming languages to include the embedding of non-procedural problems of constraint satisfaction.

The second subtheme was the construction of theorem provers that take problems expressed as theorems to be proved in the first-order predicate calculus. This line of work was motivated by the (correct) observation that the scope for representing real-world facts and situations in first-order predicate calculus is very great; and by the invention of the resolution method, a computational method for finding proofs for theorems in this calculus. There has been continuous improvement on the basic method, taking the form of proposing more powerful inference techniques, rather than the form of specific ways for programs to adapt to particular problems. The very strength of the formulation in terms of generality, namely its complete homogenization of the particular task (all tasks are seen and dealt with in the same logical formalism) turns effort away from how to exploit the particularities of special classes of tasks. But it appears that only by exploiting the particularities can significant reduction in search be achieved. From a practical point of view the only proofs produced by such problem solvers were "shallow" proofs.

Much of this line of research has been temporarily "shelved", awaiting further knowledge on how best to represent knowledge for computer processing. Problems that are essentially simple when represented in their "natural" representation appear extraordinarily complicated when translated into first-order predicate calculus. The current search for theorem provers using higher-order logics is based not on the attempt to increase the raw expressive power, so to speak, of first-order logic, but on the belief that naturalness of expression will ultimately pay off.

#### SUBGOAL 1C: High-Performance Programs that perform at near-human level in specialized areas

As the heuristic programming area matured to the point where the practitioners felt comfortable with their tools, and adventuresome in their use; as the need to explore the varieties of problems posed by the real-world was more keenly felt; and as the concern with knowledge-driven programs (to be discussed later) intensified, specific projects arose which aimed at and achieved levels of problem solving performance that equalled, and in some cases exceeded, the best human performance in the tasks being studied. The example of such a program most often cited in the Heuristic DENDRAL program, which solves the scientific induction problem of analyzing the mass spectrum of an organic molecule to

produce a hypothesis about the molecule's total structure. This is a serious and difficult problem in a relatively new area of analytical chemistry. The program's performance has been generally very competent and in "world's champion" class for certain specialized families of molecules. Similar levels of successful performance have been achieved by some of the MATHLAB programs that assist scientists in doing symbolic mathematics. The effectiveness of MATHLAB's procedures for doing symbolic integration in calculus is virtually unexcelled. Yet another example, with great potential economic significance, involves a program for planning complex organic chemical syntheses from substances available in chemical catalogs. The program is currently being used as an "intelligent assistant" in a new and complex organic synthesis.

## GOAL 2. SEMANTIC INFORMATION PROCESSING (S.I.P.)

The use of the term "semantic" above is intended to connote, in familiar terms, something like: "What is the meaning of..." or "How is that to be understood..." or "What knowledge about the world must be brought to bear to solve the particular problem that has just come up?" The research deals with the problem of extracting the meaning of: utterances in English; spoken versions of these; visual scenes; and other real-world symbolic and signal data. It aims toward the computer understanding of these as evidenced by the computer's subsequent linguistic, decision-making, question-answering, or motor behavior.

Thus, for example, we will know that our "intelligent agent" understood the meaning of the English command we spoke to it if: a) the command was in itself ambiguous; b) but was not ambiguous in context; and c) the agent performed under the appropriate interpretation and ignored the interpretation that was irrelevant in context.

In this goal of AI research, there are foci upon the encoding of knowledge about the world in symbolic expressions so that this knowledge can be manipulated by programs; and the retrieval of these symbolic expressions, as appropriate, in response to demands of various tasks. S.I.P. has sometimes been called "applied epistemology" or "knowledge engineering".

To summarize: the AI field has come increasingly to view as its main line of endeavor: knowledge representation and use, and an exploration of understanding (how symbols inside a computer, which are in themselves essentially abstract and contentless, come to acquire a meaning).

To classify all of the current work into a relatively simple set of subgoals is a formidable and hazardous undertaking. Nevertheless, here is one rough cut (stated for convenience as questions).

- A. How is the knowledge acquired, that is needed for understanding and problem solving; and how can it be most effectively used?
- B. How is knowledge of the world to be represented symbolically in the memory of a computer?
  - B1. What symbolic data structures in memory make the retrieval of this information in response to task demands easy?
- C. How is knowledge to be put at the service of programs for understanding English?
- D. How is sensory knowledge, particularly visual and speech, to be acquired and understood? How is knowledge to be applied to intelligent action of effectors, such as arms, wheels, instrument controls, etc.

Significant advances on all of these fronts have been made in the last decade. The area has a rather remarkable coherence--with individual projects threading through a number of the goals stated above (this makes excellent science and difficult exposition!)

#### GOAL 2A. Knowledge Acquisition and Deployment for Understanding and Problem Solving

The paradigm for this goal is, very generally sketched, as follows:

- a. a situation is to be described or understood; a signal input is to be interpreted; or a decision in a problem-solution path is to be made.

Examples: A speech signal is received and the question is, "What was said?" The TV camera system sends a quarter-million bits to the computer and the question is, "What is out there on that table and in what configuration?" The molecule structure-generator must choose a chemical functional group for the "active center" of the molecular structure it is trying to hypothesize, and the question is, "What does the mass spectrum indicate is the 'best guess'?"

- b. Specialized collections of facts about the various particular task domains, suitably represented in the computer memory (call these Experts) can recognize situations, analyze situations, and make decisions or take actions within the domain of their specialized knowledge.

Examples: In the CMU Hear-Say Speech Understanding System, currently the Experts that contribute to the Current Best Hypothesis are an Acoustic-Phonetic Expert, a Grammar Expert, and a Chess Expert (since chess-playing is the semantic domain of discourse). In Heuristic DENDRAL, the Experts are those that know about stability of organic molecules in general, mass spectrometer fragmentation processes in particular, nuclear magnetic resonance phenomena, etc.

For each of the sources of knowledge that can be delineated, schemes must be created for bringing that knowledge to bear at some place in the ongoing analysis or understanding process. The view is held that programs should take advantage of a wide range of knowledge, creating islands of certainty as targets of opportunity arise, and using these as anchors for further uncertainty reduction. It is an expectation that always some different aspect provides the toe-hold for making headway--that is , that unless a rather large amount of knowledge is available and ready for application, this paradigmatic scheme will not work at all.

Within this paradigm lie a number of important problems to which AI research has addressed itself:

- a. Since it is now widely recognized that detailed specific knowledge of task domains is necessary for power in problem solving programs, how is this knowledge to be imparted to, or acquired by, the programs?
  - a1. By interaction between human expert and program, made ever more smooth by careful design of interaction techniques, languages "tuned" to the task domain, flexible internal representations. The considerable effort invested by the AI community on interactive time-sharing and interactive graphic display was aimed toward this end. So is the current work on situation-action tableaux (production systems) for flexibly transmitting from expert to machine details of a body of knowledge.
  - a2. "Custom-crafting" the knowledge in a field by the painstaking day-after-day process of an AI scientist working together With an expert in another field, eliciting from that expert the theories, facts, rules, and heuristics applicable to reasoning in his field. This was the process by which Heuristic DENDRAL's "Expert" knowledge was built. It is being successfully used in AI application programs to: diagnosis of glaucoma eye disease, to treatment planning for infectious disease using antibiotics, to protein structure determination using X-ray crystallography, to organic chemical synthesis planning, to a military application involving sonar signals, perhaps to other areas, and of course to chess.

- a3. By inductive inference done by programs to extract facts, regularities, and good heuristics directly from naturally-occurring data. This is obviously the path to pursue if AI research is not to spend all of its effort, well into the 21st Century, building knowledge-bases in the various fields of human endeavor in the custom-crafted manner referred to above. The most notable successes in this area have been:

...the Meta-DENDRAL program which, for example, has discovered the mass spectrum fragmentation rules for aromatic acids from observation of numerous spectra of these molecules--rules previously not explicated by the DENDRAL chemists.

...a draw-poker playing program that inferred the heuristics of good play in the game by induction (as well as by other modes, including the aforementioned interaction with experts).

- a4. By processes of analogical reasoning, by which knowledge acquired about one area can be used to solve problems in another area if a suitable analogy can be drawn. Our human experience tells us that this approach is rich in possibilities. One successful project can be cited (and that is a limited success); a program that discovers an analogy (in full-blown detail) between a theorem-to-be-proved in modern algebra and another theorem in algebra whose proof is known. The analogy is used to pinpoint from a large set of facts those few that will indeed be relevant to proving the new theorem.

## GOAL 2B. Representation of Knowledge

The problem of representation of knowledge for AI systems is this: if the user has a fact about the world, or a problem to be stated, in what form does this become represented symbolically in the computer for immediate or later use? Three approaches are being pursued:

- B1. the approach via formal logic. As mentioned before, first-order predicate calculus was tried, but was found to be too cumbersome to represent ordinary situations and common-sense knowledge. Set theory and higher-order logics are currently under examination as better candidates to be a medium for homogeneous representation.
- B2. The ad-hoc approach. Most problem domains have a "natural"

representation that human experts use when operating in the domain. Translate that representation fairly directly for the computer, and tailor the information processes to work with it. This is the approach commonly taken, in DENDRAL, MATHLAB, in chess playing programs, visual scene analysis, and so on (almost everywhere). Though it gets the job done, it creates serious problems for the cumulation of knowledge, techniques, and programs in the science because of the inhomogeneity that arises therefrom throughout the collection of AI projects undertaken. One way out of the dilemma is to do research on the problem of translation (by program) from one ad-hoc representation into another (the so-called "shift of representation" problem). Little work has been done on this problem, except one excellent "pencil-and-paper" exercise in connection with a simple puzzle, and one subprogram in DENDRAL (the Planning Rule Generator, that translates mass spectral knowledge from its form as fragmentation processes to a form useful for pattern matching).

- B3. the approach via a "computable" semantic theory. In this approach, computational linguists attempt to analyze the full range of actions, actors, objects, and their relations, of which the common-sense world is composed; then refine and formalize these into a useable computational theory for representing facts, utterances, problems, etc. The most successful of these efforts is the Conceptual Parser (and its follow-on, MARGIE, which successfully accomplishes English paraphrase and common-sense inference).

In lieu of a tight, parsimonious computable semantic theory, other more ad-hoc systems, known as semantic-net-memory models, have developed experimenting with various sorts of actor-action-object-relation data structures. Semantic-net-memory models have a ten year history relating particularly to intelligent question-answering. Perhaps most successful of these is the HAM program which combines ideas from semantic theory, semantic-net-memory structures, and more traditional linguistic analysis (all in the context of a rather good model of human sentence comprehension, validated with dozens of careful laboratory experiments).

#### GOAL 2C. Programs for Understanding English

One can readily observe that it will be almost impossible to disentangle the skein of research on understanding natural language (English) from the coordinate efforts in representation and deployment of knowledge. Most of the state-of-the-art programs for understanding English employ, in one form or another, the basic S.I.P. paradigm outlined previously. These systems have substantial linguistic components that are highly sophisticated compared with anything done in the past. All of them incorporate

linguistic theory that has an intimate and continuous tie-in between grammar "Experts" and domain-dependent "Experts". Although the domains about which they admit discourse are still modest and discrete, they are many times richer than anything done previously. The state-of-the-art is represented by the SHRDLU program for conducting a dialogue with a simulated robot about a world of blocks, boxes, and pyramids on a table; and the Lunar Rocks program for conducting a dialogue about properties of and transformations upon NASA moon-rock samples. The SHRDLU program, for example, will carry out commands, answer questions, and generally be aware of what it was doing, so as to answer "how" and "why" questions about its behavior.

The internal structure of these systems exhibits an interesting evolution over the semantic-net-memory systems, and they appear to be a long way from the heuristic search schemes mentioned earlier. They are essentially large programs written within a programming system that provides search and matching capability. There is no factorization between a data base (i.e., semantic net) and a small set of methods that process the data base. Rather, the entire system appears to be a large collection of special purpose programs for dealing with a multitude of special cases. They give the appearance of being a highly distributed system, in which the intelligent action resides throughout the entire program.

#### GOAL 2D. Acquiring and Understanding Sensory Data.

The goal here is to discover broadly applicable methods for extracting from sensory data (chiefly visual and aural) the information that is specifically responsive to users' needs. Two classes of needs may be noted: the need to facilitate communication between man and machine; and the need to apply computers to intrinsically perceptual tasks. The former is exemplified by the desire to talk, rather than type, to computers; the latter is illustrated by the task of automatically guiding an effector on the basis of visual data. To satisfy either (or both) of these needs, it is necessary to move from well-understood problems of sensing data to much more difficult problems of interpretation.

SUBGOAL 2D1. Visual Scene Analysis. Computer-based analysis of visual scenes has its roots in work on optical character recognition (early to mid-Fifties) and by work in automatic photoreconnaissance. These tasks are essentially two-dimensional. Little is lost by disregarding dimensions of objects in a direction orthogonal to the picture plane.

AI research on scene analysis began in the early sixties with the work of Roberts on pictures of polyhedra. This work (and its intellectual descendants) differs from the earlier two-

dimensional work in two major respects: first, it explicitly considers, and capitalizes on, the three-dimensional properties of objects and their perspective representations; second, it utilizes a variety of special processing steps and decision-making criteria, in contrast to the earlier template-match/classify paradigm.

Robert's work spawned five years of intensive research on pictures of collections of polyhedra. One theme, centered on the archetypical question "Is an edge present in a given (small) region of the picture?", led to the development of edge detecting, contour following, and region finding programs. A second theme, centered on teasing out the properties of polyhedra and their representations, led to an elegant theory of permissible representations of edges and vertices, and their relations to three-dimensional polyhedra - a theory not previously discovered by projective or descriptive geometers.

Work in the polyhedral objects domain culminated in several programs capable of describing, in more or less complete detail, pictures of complicated collections of polyhedra, even taking into account shadows cast by these objects. At the same time, more complicated types of scenes began to be seriously studied. This has led to current interest in the use of color, texture, and range data, and has stimulated interest in program organizations capable of capitalizing on these multiple perceptual modalities. For example, in one paradigm perception is viewed as a problem-solving process that uses many varieties of knowledge to select perceptual operators, to guide their application to sensory data, and to evaluate the results obtained therefrom.

SUBGOAL 2D2. Speech Understanding. Research on computer recognition of speech signal data began in the Fifties with work on the recognition of isolated words. Some observations will be made here on the relation between speech understanding research and the ongoing body of AI research.

The fundamental idea driving research on speech understanding is that "recognition" is impossible (in flowing natural speech) without understanding, and that understanding is impossible without extensive knowledge about the domain of discourse. This view arises in part from the observation that ambiguities and omissions at both the acoustic and semantic level do not arise as bizarre or pathological exceptions but instead are commonplace events. Speech understanding research thus relies heavily on progress in the basic AI research problems of knowledge acquisition, representation, and deployment. This situation is unlikely to change regardless of advances in processing acoustic signals.



## GOAL 2E. Intelligent control of Effectors

This goal concerns the creation of devices and control programs for bringing about specified changes in the physical world. The effectors that have attracted the most attention have been mechanical manipulators and mobile vehicles but this has been largely a matter of experimental convenience. In principle, they could as easily have been subsystems of spaceships or manufacturing tools.

Early work in "intelligent" effectors dates back two decades, but systematic work did not begin until about 1966, at which time some progress had already been made in developing symbolic problem solving programs to control effectors. Since then there has been considerable interest in computer-controlled effectors because problems of effector control excite a set of important issues for AI research. The following is a rough characterization of the subgoals of work on effector control:

### SUBGOAL E1. Monitoring Real-World Execution of Problem

Solutions: The special touchstone of effector control research is that a problem is never "solved" until the real, physical world has been altered in a fashion that satisfies the task specification (in contrast to other problem solving programs whose responsibility ends with the symbolic presentation of a good solution). Thus, an effector control program should ideally be prepared to deal with any eventuality that affects the execution of a theoretically correct solution, be it initial misinformation, accidental dynamic effects, etc. These demands strongly influence all levels of program organization and strategy. Problem solving and execution monitoring must be made to interact intimately. The most advanced work of this type is probably the STRIPS-PLANEX system (for the control of a mobile vehicle) that can detect and gracefully recover from a wide variety of execution difficulties.

SUBGOAL E2. Modelling "Everyday" worlds: To control effectors by computer requires that the computer have adequate models of everyday situations. It has become important to model occlusion, obstruction, relative location, etc., and this has been done to the extent necessary to handle various simple manipulation and locomotion problems.

SUBGOAL E3. Planning in the face of uncertainty. Problem-solving programs for the control of effectors that operate on the physical world must be able to work routinely with incomplete and

inaccurate information. This creates a need to do research on programs that can form contingency plans, can plan to acquire information, can decide when to execute actions in the physical world, even if the plan is incomplete, and so forth. Some research of this type has been done.

SUBGOAL E4. Low-Level Control. By low-level control is meant: programs that interact more-or-less directly with the effector mechanism, and that do not engage in global planning or problem solving. Research on this topic is producing a new and potentially important branch of classical automatic control. Although little has been formalized to date, enough experience has been acquired to permit the construction of interesting demonstrations. Among the most impressive of these is an arm control program that can drive the arm in partially constrained ways; for example, the arm can be made to turn a crank by dynamically constraining the necessary degrees of freedom.

SUBGOAL E5. Hardware Development. The manipulators available in 1966, whether based on prosthetic limbs or industrial put-and-take machinery, were generally too primitive to be of long-term value for AI research. This situation fostered a fairly significant hardware development effort that produced a useful arm-hand device. Similarly, sensing devices received some development efforts. Examples of this work are newly developed optical range finders, and special tactile, force, and torque sensors.

GOAL 3. Information Processing Psychology: developing detailed scientific models of human symbolic processing behavior.

Since its inception, one focus of AI research has been the study of the symbol manipulation processes capable of explaining and predicting human behavior in a wide range of cognitive tasks. As science, the endeavor is entirely classical in intent and method, employing model construction and validation. Empirical data from well-controlled laboratory experiments is obtained from psychologists or generated by the researchers in their own laboratories. Induction from this data leads to the formulation of a symbol-processing model which purports to explain the observed phenomena. This model is given a precise form as computer programs and data structures (since the computer as a general symbol-processing device is capable of carrying out any precisely specified symbol-manipulation process; this step is entirely analogous to the model-implementation step taken by the physicist when he translates his physical model into the form of a set

of differential equations). A computer is then used to generate the complex and remote consequences of the symbol-processing postulates of the model for the particular laboratory situations and stimuli being studied. These consequences and predictions are tested against empirical data; differences are noted and analyzed; the model is refined and run again; iterations continue until a satisfactory state of agreement between model's predictions and empirical data is achieved.

From one point of view, the endeavor is to be seen as Theoretical Psychology. From another point of view, it can be seen as a systematic attempt by AI research to understand intellectual activity as it occurs in nature (i.e., in humans) so that artifacts capable of performing such intellectual activity can be constructed upon the principles discovered. The interplay between these two views has been very strong.

Information Processing Psychologists have usually chosen their problems in areas that have been of "classical" concern to Psychology, though some of these areas have been reopened to serious investigation because of the successes of the information processing approaches. The following are brief sketches of some subgoals of the effort in Information Processing Psychology.

SUBGOAL 3A. Functional reasoning. Analysis and modeling has been done for human behavior in solving logic problems, complex cryptarithmic puzzles, and chess-play problems. The models, and the predictions derived from them, are so detailed that no comparison with previous work on the psychology of problem solving is meaningful. The work is a scientific revolution, and has had a great paradigmatic and methodological impact upon Psychology. The principal innovators, Newell and Simon, have had their contributions recognized by election to the National Academy of Science; Simon was awarded the Distinguished Scientific Contribution Award of the American Psychological Association, more or less the "Nobel Prize" of Psychology.

SUBGOAL 3B. Rote Memory and Short-term Memory phenomena: Storage and retrieval processes for short-term memory. Rote memorization effects. Discrimination and association learning for verbal materials. These and related phenomena of verbal learning and memory have been studied intensely by experimental psychologists in this century. A few dozen solid empirical generalizations are known. A set of closely related information processing models is capable of explaining many of these (roughly speaking, 15-20 of the "classical" phenomena).

SUBGOAL 3C. Long-term associative memory: Associative retrieval from associative memory nets of several hundred to a few thousand

symbols. Interaction of English sentence processing and memory. The symbolic representation of knowledge (i.e., facts about the world) in memory. The work is currently very active, highly promising, and is causing a mini-revolution in thinking among psychologists who study memory.

SUBGOAL 3D. Pattern induction/concept formation. Induction of models of pattern regularities in strings of symbols. Induction of the "generating rule" from the exhibition of instances of the rule.

SUBGOAL 3E. Phenomena of neurosis. The behavior studied is neurotic symbol-processing behavior, viewed as processing distortions of otherwise "normal" linguistic and problem-solving processes. A highly successful model of paranoid behavior has been developed, incorporating some English language processing.

These examples are but pieces of a bigger picture, which looks something like this:

1. It is no surprise that Psychology has been strongly affected by the information processing concepts and tools of AI research since both sciences are concerned with the study of cognition. The magnitude of the impact is the big surprise. It is probably fair to say that the dominant paradigm currently structuring Experimental Psychology in this country is the information processing paradigm. Upon no other area of science has AI research had such a strong impact.
2. The scientific study of human thought has been accelerated greatly during the last fifteen years because of the AI impact. It is not much of an overstatement to say that the AI impact has revitalized the study of thinking by Psychology, making this scientific enterprise tractable, fruitful, and respectable.

VIEW OF THE FUTURE: What lies within a five year horizon?

An extrapolation of the research directions previously described into the future faces at least two problems. First, there are the usual uncertainties that loom because of unpredictable advances and wishful thinking. Second, the imposition by ARPA of research priorities upon the course of events that would "normally" ensue will have a large effect. Thus, the question of "what should happen" is as big a question as "what will happen."

This exposition is made difficult by the fact that the structure of the field, as outlined above in terms of Goals, will show strong confluences during the future period. Any simple presentation goal-by-goal would be misleading, and was not attempted. Instead, each identifiable focus is stated and then given an extended discussion.

The main thrusts of the Artificial Intelligence community in the next five year period will be:

1. Development of applications programs that represent and use knowledge of carefully delimited portions of the real-world for high-performance problem solving, hypothesis induction, and signal data interpretation.

The next period is likely to be a period of consolidation of AI's previous gains into meaningful real-world applications. High levels of competence in the performance of difficult tasks will be the hallmark. In addition to growing attitudes toward becoming more relevant, the AI community's current major interest in knowledge structuring and use will naturally lead it to bodies of real-world knowledge that are rich in structure and challenges. An extrapolation indicates applications to domains in science (much as the DENDRAL and MATHLAB programs were developed); and in medicine (current activity includes programs that deal with Infectious Diseases and with Glaucoma); perhaps more routine aspects of architecture (e.g., space layout and design); perhaps design in electronics (e.g., layout of IC and PC electronics, actual circuit design to functional specs); management science applications (e.g., logistics management and control, crew scheduling for aircraft fleets). The most significant application will be to computer science itself, namely the automation of many programming functions (to be discussed later). Application to some of the less routine aspects of office document processing is a likely event (discussed later). With appropriate stimulus from ARPA, or other service agencies, these application priorities could be shifted toward defense problems, particularly those related to signal processing (e.g., application to seismic or sonar signal interpretation). In such applications, interpretation of what the signal means is made in terms of knowledge about the signal-generating source and the environment in which the signal occurs. All of these applications will be characterized by careful choice of domain, careful delimiting of the extent of knowledge necessary to do the job, and close coupling with human experts to gain the knowledge necessary. None of these programs will be "general problem solvers" of the old genre. Characteristic of some of these applications will be one-line interaction with human experts, not only to "tune" the knowledge used by the program, but also to intervene in decisions for which human expertise dominates that of the program, or where the relevant knowledge has not been made explicit and formalized for computer processing.

2. The development, in particular, of that area of application involving the synthesis of computer programs (the so-called "automatic programming" problem).

The particular application of AI techniques to the task of synthesizing computer programs from imprecise and non-procedural descriptions of what a user wants a computer to do for him is the AI problem area whose time has come. This area will be the subject of a separate and detailed program plan. It is an AI application of tremendous economic, and industrial importance, since computer programming is today a major bottleneck in the application of computers to technological and business problems. What is worse, virtually no advances of substantial impact upon this problem have been made in the last decade in other areas of computer science (with the possible exception of the interactive editing, debugging, and running of programs). The automatic programming problem is, furthermore, the quintessential problem that fits the WHAT-TO-HOW characterization of the nature of the science of Artificial Intelligence. It is the meeting ground of many of the tributaries of AI research: problem solving, theorem proving, heuristic knowledge and search, understanding of English (perhaps even speech), and advanced systems work. It is an ideal problem from the viewpoint of knowledge-based systems--the main line of current AI research. The essential activity in building such systems is the extraction and formalization of knowledge of the specific task domain. In the art of programming, computer scientists are their own best experts, and for years have been engaged in formalizing what is known about programming, mathematically and in other forms. Following this line of reasoning, the programming task that may be best suited is systems programming. An example of a specific systems programming task that may be accomplishable within the period is: development of an automatic programming system that will produce operating system code for a minicomputer like the PDP11/45, in response to functional specifications for instrument control and data-handling, where the specs are given in functional terms by a scientist putting together the instrument-computer package, not his (until now inevitable) programmer.

3. The extension of current ideas about the processing and understanding of English to more extensive domains of discourse and with greater flexibility, to the point of practical front-end processors for large applications programs.

In the coming period, programs for understanding English in limited "universes of discourse" will achieve practicality, and will be made available as the linguistic interaction vehicle in some of the larger AI applications programs, e.g., the

automatic programming systems mentioned above. Since these applications programs will be domain-limited anyway, it will not be an extraordinarily difficult task to construct for them front-end processors that understand English in that domain. Since currently the field has only "demonstration programs" that exhibit (limited) understanding of English, much more research will be undertaken in these directions: examining how well current techniques extrapolate to broader domains of knowledge; developing techniques for establishing context of an interaction and maintaining that context throughout the conversation; and extending methods for drawing inferences from the continually updated context. Research on semantic theory, previously mentioned in connection with representation of knowledge, will be applied to specific problems of linguistic interaction involving actors, actions, objects, and common-sense knowledge. The area of language understanding is so rich in possibilities and implications that it is not unreasonable to consider developing a separate program plan for it within the next two years.

4. Initial exploration of office-work tasks as an area of development and application; the careful choosing and shaping of specific tasks in this enormous arena of human endeavor; and some limited applications progress on these tasks.

The AI research community has been searching for problem domains of significance to science, technology, or industry that would provide an integrating theme for the various subareas of AI work. These subareas have a considerable coherence of concepts and techniques, but the centripetal force of a real-world theme is necessary to make this coherence a practical reality. Production assembly by combinations of vision, manipulation and problem solving programs is an attempt to establish such a theme. Increasingly the feeling is growing in the AI community that the development of "intelligent assistant" programs for ordinary office work is a useful and important focus. There are two reasons for this. First, much of current AI research fits the task area well (e.g., semantic-net-memory structures, question answering programs, natural language understanding, "intelligent assistant" interaction programs, etc.). Second, the explosion of use of the ARPA network for "office work" tasks quite apart from computation (uses such as message processing, message and document filing, information retrieval from large data bases, composing and editing of documents, etc.) provides an excellent medium in which to do the work. The AI community, perhaps with a push from ARPA, has the capability to do significant work on the office automation problem in the next period. A carefully thought-through program plan will probably be the first output of the field in this area (should be organized and completed within the next two years), followed by initial exploratory

ventures along the lines laid down in the plan. Again, as with all the knowledge-based systems of this decade, the specific tasks worked upon will of necessity be carefully delimited. The general "intelligent office assistant" is well beyond the horizon, but specific assistant-programs for handling some of the office-work flow of information on the ARPA network can be realized within five years.

5. Intensive developmental work on the speech-understanding problem.
6. Expansion of computer vision research to: knowledge-based program organizations; development of a repertoire of low-level perceptual operators for color, range and texture, and exploitation of these modalities; first practical applications of scene analysis to selected tasks in industrial and biomedical settings; and use of interactive scene analysis for both research and application purposes.

Scene analysis programs consist of a combination of sensing-and-measuring primitive perceptual operators (like line-finders) and higher-level knowledge-based procedures (like line-proposers). Because of general awareness of the limitations of current primitive operators (at least as they are applied to monochrome pictures), the research will place increased emphasis on the acquisition and low-level analysis of color and range data. Higher level procedures will use knowledge of: three-dimensional properties of objects other than polyhedral objects; perceptual properties of objects; many varieties of contextual constraints among objects; and properties of the primitive operators (like computational cost, reliability, and domain of applicability). Practical applications will probably focus on industrial tasks like work-piece identification and location, inspection, and manipulator control. The scene analysis research issues in these applications may turn out to be pedestrian, but concerns about cost, reliability, and reprogrammability will become prominent. Biomedical scene analysis problems will continue to stimulate research; application to medical mass-screening tasks may occur. Interactive scene analysis will be an important focus. In research settings, interactive scene analysis will be used to construct large scene-analysis systems through the incremental accumulation of knowledge; in application settings it will be used to achieve flexible scene analysis systems that can be easily "re-programmed" by users who are not computer scientists.

7. Expansion of arm-hand effector technology and associated program control, with some practical applications of simple forms of this technology in industrial settings.



There will be considerable activity in the transfer of ARPA-initiated work on effector control to industrial settings. Hardware realizations of a rich variety of mechanical effectors, with their tactile, force, and torque sensors, will appear. Visual feedback in controlling effectors will be a feature of many of the applications. Basic research on the hardware and software technology of effector control will continue, if support from ARPA or other agencies is forthcoming. More broadly-based research on effector control is likely to be stimulated by the appearance of relatively inexpensive experimental hardware. Researchers who are currently unable to develop one-of-a-kind devices because of their cost will enter the field.

8. Expanded basic research on acquisition, deployment and representation of knowledge to support knowledge-based systems development.

Though the main thrust of AI research is in the direction of knowledge-based programs, the fundamental research support for this thrust is currently thin. This is a critical "bottleneck" area of the science, since (as was pointed out earlier) it is inconceivable that the AI field will proceed from one knowledge-based program to the next painstakingly custom-crafting the knowledge/expertise necessary for high levels of performance by the programs. In the next period, the following kinds of fundamental explorations must be pursued and strongly encouraged:

- a. Additional case-study programs of hypothesis discovery and theory formation (i.e., induction programs) in domains of knowledge that are reasonably rich and complex. It is essential for the science to see some more examples that discover regularities in empirical data, and generalize over these to form sets of rules that can explain the data and predict future states. It is likely that only after more case-studies are available will AI researchers be able to organize, unify and refine their ideas concerning computer-assisted induction of knowledge.

- b. Development of interactive interrogative techniques, coupling a program to a human expert, by means of which the program systematically elicits from the expert particular facts, useful heuristics, and generalizations (or models) in the domain of the human's expertise. Again, specific case-studies are desirable. Their development need not await the arrival of English language understanding programs to facilitate the interaction and interrogation. (Stylized languages designed for the specific case-study domains will serve for now.)

c. Exploration of a variety of methods for bringing together disparate bodies of knowledge held by a program to assist in the solution of specific problems that the program is called upon to solve. The nature of this problem was discussed earlier under Goal 2A. If there are to be a number of Experts (i.e., specialized knowledge bases) interacting in the solution of a problem, how should their interaction be arranged? Is there an Executive Program "in charge" of sequencing the activity of the Experts? If so, what is the nature of the Executive Program's knowledge about each Expert, and the appropriateness of calling that Expert to assist at a specific point in the process? Should the Experts be relatively independent, each with its own situation-recognizer to trigger its activity? These particular questions are posed here not in an effort to characterize the problem completely (or even adequately), but to give the flavor of the experimental inquiry that needs to be pursued in the coming period - a period in which major AI programming efforts will be directed toward knowledge-based systems with multiple sources of knowledge.

d. Theoretical and experimental studies of representation of knowledge. This basic and difficult problem is not one that is likely to have a "solution" in a five year period. Theoretical studies will continue to search for a logical calculus in terms of which to formalize and store knowledge in a fairly "natural" way, and for logical processors that will compute efficiently within this formalism. Experimental studies will attempt to deal with the usual nonhomogeneity of representation among different bodies of knowledge directly, by programming translations of representations from one "natural" representation to another as necessary in those situations requiring communication between Experts for joint problem-solving.

9. Continuing basic research on various mathematical-logical problems such as formal models for heuristic search, theorem proving methods, and mathematical theory of computation.

Because heuristic search has been a central theme of AI problem solving research, it is likely that attempts at mathematical formulation and analysis of heuristic search methods will continue. No existing research thrusts indicate that this work should have high priority at this time. However, the situation is unstable in the sense that a few key results (e.g., new theorems or, more likely, new formulations of heuristic search) might cause a rush of activity along lines of formal analysis.

A similar situation attends theorem-proving research. There are currently no critical ideas acting as a forcing function, but nonetheless the problem appears to some scientists to be central for progress in the long run. In their view, to state

that a computer can be used as a "symbolic inference engine" is equivalent to saying that it is a "logic engine"; and what makes a "logic engine" turn over is a theorem prover over the domain of some logical calculus. The search for appropriate logical calculi and associated theorem provers will therefore continue.

The work in mathematical theory of computation has been peripheral to the AI mainstream, but recently has been gaining momentum and importance, and will enter the mainstream as basic research for automatic programming efforts. To write programs capable of synthesizing programs obviously requires a thorough understanding of the nature of programs. One kind of understanding is gained by formal description and mathematical analysis (the kind of understanding we take so much for granted in some physical sciences and engineering). To the extent that useful formal descriptions of how programs are put together and what programs do can be discovered; and to the extent that powerful theorems can be proved within the formalism; the work on mathematical theory of computation could aid significantly in the practical work of constructing automatic program synthesizers and verifiers. Thus, there are noteworthy "breakthrough" possibilities in this area.

A prediction of the most likely course of events in these tasks of formal analysis is that they will be low-key, low cost, high risk/high payoff.

10. Continuing research on modeling of human cognitive processes using information processing techniques.

At the interface between AI and the psychology of human perception and thought, the research tempo has been increasing for some time. In the coming period it is likely that new methodology, new conceptual insights, and new models will have a continuing dramatic impact on Psychology. The feedback to ongoing AI research will continue to be important, particularly in the areas of perception and memory. The principal developments are likely to be these:

- a. Methodological: analysis by program of the thinking-aloud protocols of humans solving complex problems (i.e., "data reduction" that requires some language understanding and complex inductive inference), resulting in a speed-up in this critical empirical procedure of perhaps a factor of 100. A typical complete protocol analysis of human data in a puzzle-solving task currently takes, without computer assistance, 100 hours.

- b. Short-term memory. The processes of human short-term memory will be so well modeled and understood as a result of

research in this period that the topic will cease to be of major theoretical interest to psychologists.

c. Long-term memory. A very good model of human long-term associative memory will be developed. The program which realizes this model will be given a great deal of "garden variety" knowledge of the everyday world, as the basis for empirical testing. Such a model will undoubtedly prove to be an important subsystem in larger programs that attempt language understanding in contexts involving common-sense knowledge. Only the beginnings of such a memory model exist today.

d. Visual perception. The most important impact of AI on Psychology in the coming period may be the initial formulation on an information processing theory of human visual perception of common 3-D forms, along the lines of the visual processing concepts and operators developed by AI vision research. AI vision research stands on the threshold of Psychology awaiting an intellectual push like the one given to problem solving in late Fifties. If the push is made, and is successful, it will noticeably dent the theory of visual perception in five years and totally capture it within ten years.

## APPENDIX B

## Justification for Storage Augmentation - July 1974

The following is the text of a proposal submitted to the AIM Executive Committee and the NIH-BRB at the end of July 1974 for augmenting file storage, memory, and swapping storage capacities for the SUMEX-AIM resource. The committee approved the proposal and, as discussed in the text of the report, we have implemented the file space and memory additions to good effect. The swapping storage augmentation has been pending until we felt a clear demonstration of need existed.

Based on system performance measurements over the past several months, we have come to the belief that whereas we are at the capacity of the swapping storage now under peak load, there may be a software remedy which will delay the need in this area. SUMEX CPU capacity has become a rather more critical resource at this time with the growing community of users. We are currently formulating an additional plan to augment the system in terms of processor power. This will be submitted for review during the next grant year (03).

RECOMMENDED SYSTEM STORAGE AUGMENTATIONS  
WITHIN FIRST YEAR BUDGET  
JULY 29, 1974

The initial SUMEX computer configuration plan, approved by the AIM Executive Committee in November 1973, was a compromise between the technical demands of establishing an effective community AI computing facility and the budget constraints imposed by Council. Within the projected budget at that time, we attempted to balance the configuration in terms of available file space, core size, and swapping storage.

As discussed at earlier Executive Committee meetings, ARPA has found it necessary over the past 6 months to reconsider its policies as they relate to ARPANET expansion and use by non-DOD agencies. The result of these deliberations has been a decision in early July by ARPA that SUMEX can become a Very Distant Host (VDH) on the ARPANET rather than a new TIP node as initially planned. We have revised the earlier network plan to implement a VDH interface and to augment the interim line scanner capacity to handle local terminals (previously to be handled by the TIP). We are also in the process of interfacing to the TYMNET in order to provide low bandwidth terminal support on a broader geographical and administrative basis than is afforded by the ARPANET at the present time.

Some reductions in first year costs have resulted from the reconfiguration and delays in implementing the ARPANET connection. These include delayed project staffing, delayed operational status, and reduced communications fees as well as the inherently lower cost of the VDH connection. The overall reductions amount to approximately \$148,000 and afford the opportunity to reconsider other aspects of the machine configuration to give a larger capacity to better meet the needs of the AIM community.

Whereas the SUMEX facility is just coming to a fully operational state, we can project a number of areas where augmentation would be of benefit to system performance. These projections are based on observations of current SUMEX utilization as well as experiments on a KA-TENEX system at the Institute for Mathematical Studies in the Social Sciences (IMSSS). The IMSSS machine allows a more parametric measurement of performance sensitivity to hardware changes because it has a larger configuration from which the effects of reducing various component capacities can be observed. The following summarizes these recommendations.

## FILE SPACE

Even in these early stages of SUMEX operation, it has become clear that the file system capacity will be a limiting factor to AIM community expansion. This derives from the interactive nature of the TENEX system making on-line files essential, the large files involved in AI program images, and the large data files currently in use and expected increasingly as data base-oriented AIM projects are identified. The capacity of the current file system is not yet fully utilized and we have issued only verbal requests to economize on file space. However, the trend toward early consumption of the file capacity is clear as summarized by recent file utilization statistics.

Out of a total of 81,200 available pages (4 RP-03 disk drives), the following are averages of the space in use including all system and user directories:

Mid-June	47,500 pages
Late June/early July	53,000 pages
Mid-July	52,000 pages
Late July	52,000 pages

We have developed a policy statement on file space allocation and control which is attached. In this policy, current data on disk requirements for various aspects of the system and user projects are integrated to allocate the overall available space (81,200 pages):

I. TENEX/AIM SYSTEM (common to both SUMEX-SUMC and -AIM)

Operating Monitor	5,000 pages	
Supporting Direct. (lang., lib., etc.)	10,000 pages	
AIM management and SUMEX staff	10,000 pages	
File system reserve for temporary overflows	6,200 pages	
	TOTAL	31,200 pages

II. SUMEX-SUMC Users

TOTAL	25,000 pages
-------	--------------

III. SUMEX-AIM Users

TOTAL	25,000 pages
	-----
	81,200 pages

Among the initial SUMEX-SUMC projects (DENDRAL, Protein Structure Modelling, MYCIN, and various pilot efforts) approximately 17,500 pages are in use. On the SUMEX-AIM side only 8,000 pages are allocated because delays in network connections have precluded Dr. Amarel's and Dr. Colby's groups from actively using the system.

Based on these data, we recommend adding 4 more drives (81,200 pages - this is also the limit of the number of drives which can be put on the existing controller) to augment the SUMEX-AIM component of the file system. This would provide room for an additional 8-16 projects at 5,000-10,000 pages per project. At \$13,000 (plus tax) per drive, the total cost for this augmentation would be \$55,120.



## MEMORY AND SWAPPING STORAGE

The operational status of the SUMEX KI-TENEX system has been approaching "routine" since May for the local community primarily. Over this period we have begun to collect statistics on the performance of the system but note that swapping is implemented on a provisional, inherently inefficient basis on the moving head file system disks. A sample of these data is shown in Figure 1. During the prime time shown, the system load was 10-14 jobs including 2 or 3 LISP users and miscellaneous EXEC, editor, and private program jobs. Plots are shown in Figure 1 of the percent time allocated to running user programs and the percent time consumed in system overhead (waiting for pages to be swapped in and out to make a program runnable, managing core allocations, and handling page fault traps). It is significant to note that the overhead consumes on the average about 35% of the machine under this load and in excess of 60% at times. This is predominantly a result of I/O waits on the relatively slow disks used for swapping. During this period, the maximum demand for swapping storage was 1750 pages.

A dramatic improvement in efficiency is expected when our permanent fixed head swapping device is installed in August, but these data raise obvious questions about the system capacity which will be allocatable to additional user projects. In conjunction with Mr. Rainer Schulz of the Stanford IMSSS facility, we have collected a preliminary set of data illustrating the relationship between system overhead and hardware configuration. The IMSSS KA-TENEX facility was used because they have a total configuration of 256 K words of memory and a large swapping drum in operation so that by limiting each of these parameters, we could evaluate the overhead under a "standard" load. The results of this experiment are shown in Table 1.

At present, the SUMEX machine is operating in a configuration similar to box 5 in Table 1 and with the installation of the swapping device will operate somewhere between boxes 1 and 3. (Note that the amount of virtual address space overflowing the "drum" determines the relationship of box 3 between boxes 1 and 5). The interaction between overhead, configuration, and job mix is complex. Witness for example, some data not shown in the Table. By adding 2 100 page jobs to the 4 200 page jobs in boxes 3 and 4, the overhead in box 3 is lowered while that in box 4 is raised. Nevertheless, several general relative trends can be noted. Increasing the speed of swapping storage reduces system overhead by reducing the I/O wait time for moving pages in and out of memory. Increasing memory size also reduces the overhead by allowing more working sets to be resident simultaneously thereby giving more candidate jobs to be run while waiting for pages to be swapped for other jobs.

It must be noted that the jobs run in this test simulate

the effects of simultaneous very large jobs. In general there will be a spectrum of job sizes which will tend to reduce the overhead in all configurations (more working sets resident). On the other hand, the overhead estimates for swapping off of moving head disks are low because no data files were in use during the test thereby necessitating fewer time-consuming head seeks than would be encountered normally. Also the test programs addressed their arrays sequentially so that large blocks of pages would tend to be sequentially resident on disk. Thus in swapping programs in and out, less seeking would be required than normal.

From these estimates of relative system overhead as a function of configuration, it is clear that substantial gains can be made by adding memory to the system and by guaranteeing enough capacity so that paging occurs off a fast, fixed head device. This relative overhead can be reduced from something in excess of 30% (box 3) to something in excess of 11% (box 4) by adding memory and from greater than 11% to about 8% by adding more swapping storage. The improvement in efficiency by adding swapping storage would in fact be more than is apparent from the above data, taking into account the additional inefficiencies involved in more randomized disk seeks. Note that on the day data were taken for Figure 1, the maximum swapping space in use was 1750 pages. The fixed head swapping device we are getting will have a capacity of 2600 pages. Thus, in normal operating circumstances the probability that swapping storage will overflow to the slower moving head disk is real.

Even for a 100% efficient system, the number of users which can effectively be accommodated is limited by the response time for each user given roughly by a subdivision of the CPU capacity between the total number of users. It is very hard to pin down this number at present because it will depend on the nature of the jobs in execution. In the grossest terms, we might expect one limiting complement of users to be on the order of 5-10 LISP jobs (300-400 pages each) and 20 smaller jobs (50-100 pages each) for a total of something over 4000 pages of address space in use. This would clearly overflow the 2600 page swapping device.

For the above reasons, even though the firm limits of the current machine configuration have not been reached by existing user community demands, augmentations of the system memory and swapping storage would be beneficial to the AIM mission in allowing a larger community of projects to participate. Within the first year budget allocation, 64 K words of fast memory can be added (\$50,000 plus tax) and the swapping storage doubled (\$37,600 plus tax). Based on the relative data in Table 1, these additions, while costing about 10% of the overall facility, may free up approximately 20% of the machine capacity from overhead. This extra capacity is significant in terms of added AIM user

support. We therefore recommend these augmentations in addition to the file system expansion discussed previously.

The total augmentations can be accommodated within the expected first year budget underrun:

File storage	\$55,120
Memory	\$53,000
Swapping storage	\$39,850
	-----
TOTAL	\$147,970

FIGURE 1

"WATCH" DATA TAKEN 7/24/74 AT 15 MINUTE INTERVALS

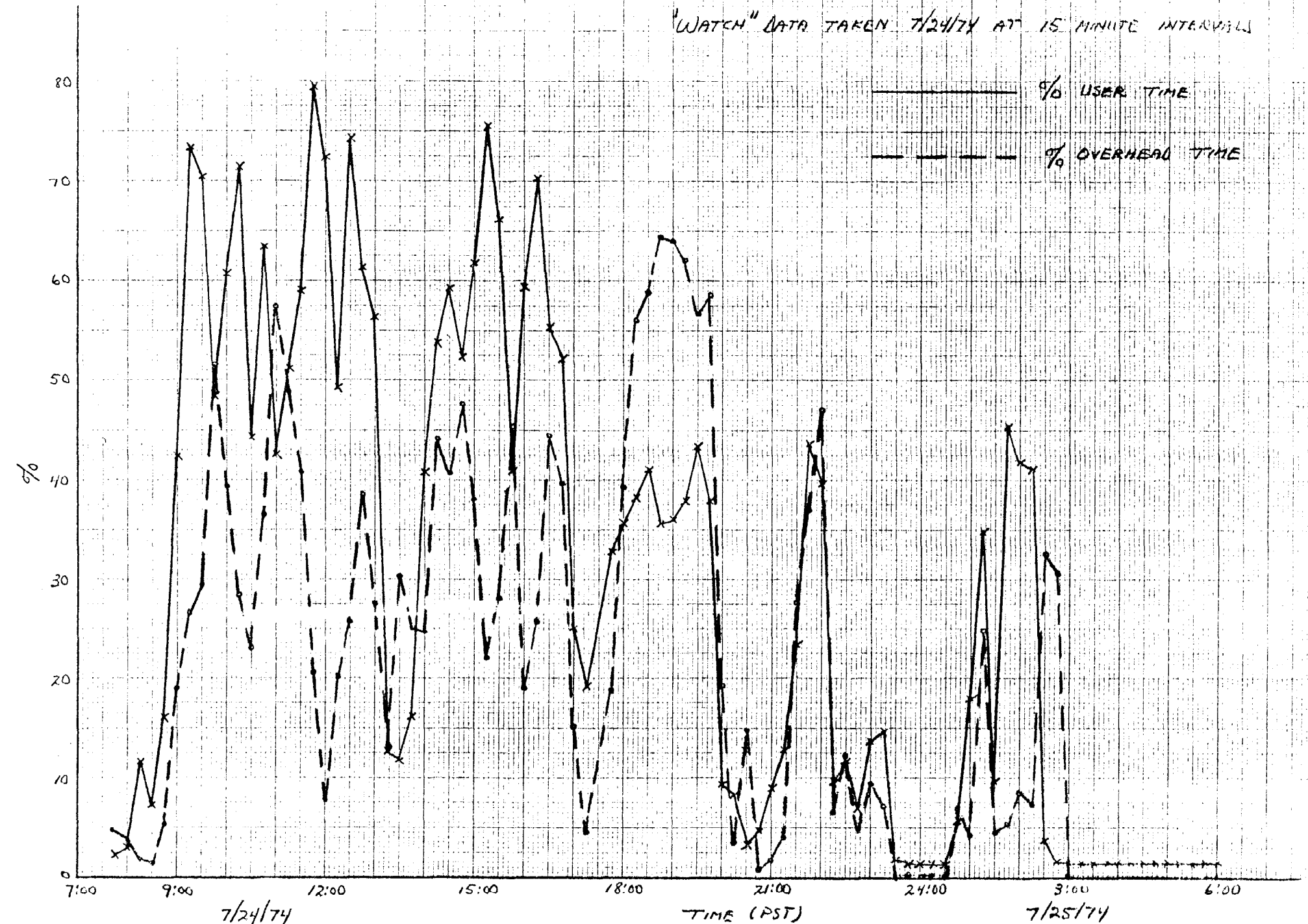


Table 1

System Overhead as a Function of Configuration

	Memory	
	196 K	256 K
All "drum" (fixed head)	1 9 x 100 pages: 6% 4 x 200 pages: 16%	2 9 x 100 pages: 4% 4 x 200 pages: 8% 4 x 300 pages: 24%
Part "drum" / Part "disk" **	3 9 x 100 pages: (11%)* 4 x 200 pages: (33%)*	4 9 x 100 pages: (5%)* 4 x 200 pages: 11%
All "disk" (moving head)	5 9 x 100 pages: 17% 4 X 200 pages: 51%	6 9 x 100 pages: 6% 4 x 200 pages: 15%

\* Estimated by interpolation because actual measurements were not available

\*\* The drum space was limited to 450 pages with any overflow moving to disk

## APPENDIX C

## Assessment of System Responsiveness Under Load

The reports from the individual projects in the SUMEX-AIM community (see Section IV, for example page 80 and page 93) suggest that the system loading is subjectively approaching saturation and express concern over the ability to work during prime time and to be able to have physicians use the interactive programs with enough responsiveness so that their frustration does not go so high as to discourage them from further use. In addition to these comments volunteered in the reports, we have asked other users as well to gauge their subjective impressions of responsiveness and those of their medical collaborators against load average measurements. Most express concern about being too precise in their judgements but generally agree that very noticeable response degradations set in when the load average gets above about 4 or 5 and that responsiveness deteriorates increasingly (non-linearly) above that. In several instances users expressed concern about the long time needed to load (not just execute) large programs when the system is heavily used. Still others get frustrated when they seemingly get no attention from the CPU at all during some intervals of time. A table relating subjective feeling to load average as submitted by one user is reproduced below as being fairly typical of the reactions received.

" LOAD AVERAGE -----	PERSONAL FEELING -----
< 1.0	Heaven. Echo great, no delay. interactive programs are. When response is not immediate, you know something important is happening.
1.0 - 2.0	Not bad at all. Slowdown is perceptible, but easily tolerable.
2.0 - 3.0	Livable. Echos are delayed by now, and impatience begins on waiting for timeout to catch up. Simple problems in a LISP job are doable, but larger ones are getting long.
3.0 - 5.0	Only editing proceeds with ease. I do practical problems only if they are extremely important (this gives me an increasingly small window these days, as the load average is frequently in this area).
> 5.0	Interactive programs aren't very, and aren't at all much above 5.0. The most trivial of demos is painful, as the response (echo) can be seconds, and the program response on even a simple task is long. Even FORTRAN programs bog down badly here. "

It is difficult to precisely quantitate the subjective aspects of response time, relating user frustrations to objective loading measures, because of wide variations between user personalities and the interactive quality of various programs. Typically, the more intimately interactive a program is the more easily user frustrations are built up with response delays. The fairly commonly expressed break point in the responsiveness versus load average curve at a load average of 4 is understandable as this is about the number of runnable working sets which can be kept in memory at a given time (user memory is about 380 pages which will hold 4 working sets averaging 95 pages each). With fewer than 4 or 5 active jobs, each job gets an aliquot of CPU time periodically on a fairly continuous basis. The main effect causing slowness is that a given user gets only about one quarter or one fifth of the PDP-10. Above this level, some jobs have to be swapped out and get no CPU time until when they are again brought in. From the user's point of view, the system sits idle on his job for an interval of time and then he gets a interval of attention. These intervals of inactivity produce particular frustration for some people as indicated in some of the comments. The problem is especially acute for large jobs (LISP programs) because they are more likely to have to be swapped out. Smaller jobs (like text editors, some language processors, and utility programs) tend to fit into the "cracks" when memory is allocated and hence see better service. This situation is reflected in the user comments as well.

An additional factor under very heavy loads at present is the overflow of paging traffic to the moving head disks which occurs occasionally as we do not have the more intelligent fixed head storage manager operating yet. When swapping moves to the disks, two things happen. First, the time to move a page in or out increases by a factor of 4 or 5 and second, the disk channel is tied up more often causing contention between swapping and other disk input/output activities. This may cause particularly long load times for very large programs. This latter problem should be remedied by software changes to allocate fixed head space to active pages and to move dormant ones to moving head storage.

From a system design point of view, the remaining response problems are affected by two system resources; memory size and CPU power. Adding more memory would allow more of the runnable jobs to be resident at a given time, thereby giving more continuous attention to more jobs. Increased memory would move the break point in the response curve caused by swapping toward a higher load average. Already though (based on the above comments), at a load average of 2-3 the response is degrading. Since we are running quite efficiently now (15-20% overhead), allowing more jobs to be in core to compete for CPU power would not effectively solve the response problem. It would have the beneficial effect of "linearizing" the degradation at loads above 4.

A second approach would be to increase the inherent processor speed so that jobs would get finished sooner and leave the runnable state more quickly because they are actually completed rather than having their aliquot timed out. This would reduce the load average with the same number of users because jobs would be waiting for user input more of the time. As noted in the report, the experience in upgrading the IMSSS KA-10 to a KI-10 dropped the load average from around 15 to about 5-10. Added CPU power would not change the location of the swapping break on the response vs load average curve, it reduces the load average on the existing curve. It also has the effect of reducing the number of swapping cycles a job has to go through to complete execution when swapping does set in.

In the long term both augmentations would likely be desirable. There is an optimum balance between CPU power and system memory size. That balance is reached for interactive jobs when the memory holds enough runnable jobs that when the processor is distributed over all of these jobs, just acceptable user response is achieved. Additional jobs would cause swapping which incurs a more steeply rising response penalty per added job than if more memory were available, but the response would become unacceptable in either case.

On the basis of current data and since only one of these resources could be upgraded within the present budget, we feel the CPU augmentation is the better choice. It would alleviate the present bottleneck because it improves the user-perceived response time by improving the actual running time of his job. In addition at lower load averages, it would allow more complex programs to become interactive because they would run faster.



For these reasons as the SUMEX-AIM system becomes progressively more heavily loaded with the addition of new users and collaborators, we feel the CPU will be the next most critical resource of the facility. We are currently working on a preliminary plan (see page 51) proposing to allocate available funds within the council-approved award levels to up-grade the present KI-10 CPU. We expect this plan to be refined in the next few months at which time we will submit it for AIM Executive Committee and NIH review.

## APPENDIX D

## PDP-11 SAIL Design Summary

## SAILEX --- SAIL EXperimental compiler

The Portability Problem  
-----

The number and availability of computers has expanded greatly in recent years. Accompanying this expansion has been a proliferation of programming languages and software systems. Usually, each computer has its own assembly language, and supports one or more high-level languages. The software is written in assembly language, and hence can execute only on the host computer. The many common functions performed by systems software are often obscured by implementations peculiar to a given computer. Assemblers, compilers, text editors, and linkers require separate manuals for each computer. Languages such as FORTRAN, ALGOL and BASIC have restrictions, extensions, and usually many undocumented idiosyncrasies. Operating systems have widely differing capabilities, though the underlying computers may not be so diverse. Programs often have dependencies on the available environment, or employ peculiarities of a particular computer model.

Such computer-specific programming limits a program's portability, i.e. its ability to execute on more than one computer. Some programs are unlikely candidates for portability, e.g. a program which communicates with a one-of-a-kind device. Many programs, however, could be written so that only minimal changes, if any, would be needed to allow execution on another computer. Since assembly languages are designed for specific computers, portable programs must be written in a higher level language, i.e. a language which does not concern itself with the physical makeup of the computer. Machine-independent languages focus attention on the problem to be solved, rather than the implementation of the solution.

The  $M \times N$  problem involves  $M$  languages which are to be executable on any of  $N$  computers. Each language could have a compiler for each machine, resulting in  $M * N$  compilers. Alternately, there could be  $M$  compilers which translate the languages into a single intermediate code, and  $N$  compilers which translate the intermediate code into target code, for a total of  $M + N$  compilers. Further reduction in the number of compilers depends on similarities in the languages or target computers. In what follows we shall be concerned with the compilation of a single language for many computers, a reduction of the  $M \times N$  problem to the  $1 \times N$  problem. A distinction should be made between "language portability" and "program portability." By language portability we mean a language which can be compiled into code for many computers. Program portability is the ability of a particular program to be executed on many computers.

## Language Portability

---

A number of new developments in computing will have a great impact on language portability. The growing literature on programming techniques is making programmers more aware of general techniques, and more willing to build on the work of others. Computer networks will lead to large-scale sharing of programs, and the desire to execute programs on one computer which were developed on another. Experience on the various computers available over a network should lead to increased awareness of machine-dependent aspects of a program, and an effort to write programs usable by other computers on the network.

Memory capacity, available peripherals, even instruction sets sometimes vary little from computer to computer. Computer families with differing options between models are becoming commonplace. A single language could be compiled into code tailor-made for a particular model, so that programs written in the language could be independent of the target machine. Internal organizations are often comparable, as evidenced by the common general register --- base/displacement architecture. Increases in speed and memory help alleviate inefficiencies caused by not using a language specifically written for a computer.

Thus the feasibility of and motivation for language portability seem well established. There is a need for easy compilation of programs for different computers. Each compilation should fit the program to the target machine, making as much use as possible of the target instruction set. The compilation should not spend a great deal of time determining the characteristics of the target machine. Decisions concerning code-generation strategy should be made once, then incorporated into the compiler. The development of compilers for new machines should be an integral part of the compiler system, with as much work as possible done automatically. A simple yet flexible manner of specifying target machines should be available. The compiler itself should be written in the language being implemented, so that it can execute on computers for which it can generate code.

## Program Portability

---

Program portability is more difficult to obtain than language portability. A program written in a portable language can be compiled into code for many computers, but execution of the code may produce various results on the different computers. For example, such a program could have dependencies on word size, numeric representation, character representation, memory size, addressability, or the file system. Mathematical routines implemented on computers with differing numeric precision are difficult to port without rather elaborate precautions. Programs which must interact with the operating system, or otherwise "liberate" themselves from the programming language, must be given special attention to minimize such machine-dependence.

Portability should be considered in the initial design of a program. A little inefficiency may be acceptable to make a program portable. The resulting implementation is often more efficient, more flexible, and more easily understood and maintained. When machine-dependence is felt necessary, it should be isolated and well documented as such. Many programs which may seem to be machine-dependent can be made portable by judicious design. For example, text editors often have many dependencies on the operating system and file system. Yet the operating and file systems, and the editors, may have very similar capabilities. Thus the dependencies are not inherent to text editing, and an editor capable of running on all computers could be designed with little loss in efficiency.

#### SAILEX --- A Machine-independent Compiling System

---

A machine-independent compiling system is being developed for a subset of the language SAIL, the Stanford Artificial Intelligence Language. SAIL is described in MEMO AIM-204 of the Stanford Artificial Intelligence Laboratory, Stanford University. Changes made to SAIL for use on the TENEX operating system are described in TENEX SAIL, Technical Report No. 248 of IMSSS (Institute for Mathematical Studies in the Social Sciences), Stanford University.

SAIL was originally designed for implementation on a PDP-10 computer, but much of the language is machine-independent. Some parts considered too machine-dependent have been eliminated from SAILEX, and some features have been added to enhance the power of SAIL (e.g. double precision). SAIL is certainly not an ideal language for portable programming, but has some characteristics which make it a suitable experimental language for this purpose. A compiler already exists for SAIL, so that the SAILEX system (which is written in SAIL) can be compiled and executed without hand translation to some other implemented language. The language is powerful enough to give a good test of the feasibility of compiling for many computers. A rather large number of users already exist, and there is great interest in using SAIL on machines other than the PDP-10.

A means of specifying the semantics of a target machine has been designed to meet the previously discussed criteria. This specification includes:

1. External procedures to be available without declaration. For example, double precision numerical routines may be available on some machines.
2. Registers, and register classes. Examples of register classes are integer registers, floating point registers, and index registers.

3. Information needed for storage allocation. For example, the number of addresses per integer.
4. Switches governing operation of the resulting compiler. For example, can the symbol table be kept in memory; is readable intermediate code desired.
5. Code generators. A language has been designed for specifying code generation (the compiler outputs assembly language, not binary code). This language gives the code generators the appearance of the target code being specified. The full power of SAIL can be used to write the code generators, but due to the built-in capabilities of the generator language, such power will probably never be necessary.

The compiler makes almost no assumptions about the target machine. Extensive procedures for manipulating registers and their contents are available, but are used only if invoked by a generator. Much of the work of code generation has been found to be machine-independent; only the specifics of a target machine need be determined. The generators are responsible for requesting loading of registers for instructions which require register addressing, most local optimization, choice of instruction sequence, addressing format, and allocation of storage. The compiler does all the bookkeeping tasks such as remembering what is in the registers; loading, storing, clearing and marking registers as requested; searching for the "optimal" free register; remembering what operands have been used, for later allocation; parameter passing schemes; symbol table maintenance; and file manipulation.

The semantic specification of a target machine is used to create a compiler specifically for that machine. Extensive conditional compilation insures that those parts of the compiler not necessary for a particular machine will not be present. Compilers have been created for the PDP-11/45, PDP-11/40, PDP-10, IBM-SYSTEM/360, VARIAN-6201, and NOVA. More computers are being considered, e.g. CDC family, INTERDATA, BURROUGHS family, DATAPOINT, and SIGMA. The system will continually be generalized as these machines are examined.

Specification of a new machine without radical departures from all previously specified should be straightforward. The PDP-11/40 was specified in two hours, but the specification of the PDP-11/45, which had already been specified, was used as a template. The VARIAN machine took two days, but this included learning the instruction set (the code has not been carefully checked). In general, a poor implementation can be produced very quickly, a compiler can be generated, and the resulting code checked. This indicates how the code can be improved, and the process starts again. A compiler can be completely created from the semantic specification in a matter of minutes (depending, of course, on the speed of the computer being used). A manual describing how to specify a target machine, with extensive examples from those already described, will be produced. Familiarity with SAIL and the specification manual should be

sufficient to produce a compiler for a new target machine with little trouble.

The code produced is not globally optimized, but otherwise rather good. For example, SAILEX appears to produce 20-30% less code for the PDP-10 than the currently available SAIL compiler. Comparison with the code generated by the FORTRAN compiler on the PDP-11 and an "equivalent" SAIL program indicates a reduction in size of about 30%. Such measurements are not precise, and are given only to indicate that SAILEX does not output inefficient code.

Care has been taken to insure that the SAIL compiler is portable. By compiling the compiler, a SAIL compiler can be created which runs on the target machine (a runtime system must be written for the target machine). This has been done for the PDP-11/45, so that the SAIL compiler is available on a PDP-11/45 (also on a PDP-10). (The SAILEX compiler, together with the runtime system and symbol table space, takes about 20K words on the PDP-11/45 (the runtime system is a library). About 2K more words are needed for string space.)

Future areas to be considered are:

1. Final testing of the PDP-11/45 runtime system, and preparation for its export to other sites.
2. Specification of more machines, and resulting generalization of the code generation scheme.
3. Removal of any "hidden" machine dependencies in the compiler system.
4. Implementation of more of SAIL (e.g. full macro facility, records and references).
5. Machine independent global code optimization. The compiler has been designed to facilitate code optimization within a variable size window about the intermediate code. This is not yet done.
6. Design of a machine independent runtime system. The PDP-11/45 runtime system is written in PDP-11/45 assembly language for execution under DOS, version 9. Much of the design could in fact be abstracted from this setting. This abstraction might be viewed as a blueprint from which runtime systems for other machines could be developed, or it might be possible to actually write most of the system in SAIL, and directly export it.

## APPENDIX E

## Subsystems and Documentation Directories

Nancy Smith  
 December 1974  
 (updated April 1975)

The sources of available documentation for these programs will be abbreviated as follows:

TUG Tenex User's Guide (1975 edition)  
 DUH DEC Users Handbook  
 DAL DEC Assembly Language Handbook  
 DML DEC Mathematical Languages Handbook  
 HC a hard-copy manual for the language  
 OL on-line documentation which can be found by  
 @DIR <DOC>programname.\* . The following extensions are  
 used on the <DOC> directory:

.MANUAL	complete usually fairly long manual
.HELP or .HLP	shorter summary, list of commands, etc.
.SUPPLEMENT	on-line supplement to hard-copy doc
.UPDATE	list of updates by date
.SAMPLE	sample program or output

See <DOC>A-LIST-OF-ALL-AVAILABLE-DOCUMENTS.INFO for complete details on these documents including where and how to order them.

Many of the major programs also have a <BULLETINS>programname.BBD file where messages about new developments, bugs, hints for using the program etc. are sent. These <BULLETINS> files can be read by any of the mail reading programs (READMAIL, RD, or BANANARD).

New programs or new versions of old programs will be put on <NEWSYS> for a trial period. The file <NEWSYS>NEW-SYSTEMS.INFO which is a message file will have a message about each program available. The doc for these new programs will also be kept on the <NEWSYS> directory. These new programs will not be included in the list of programs given here.

The EXEC command @HELP prints a file with general help information.

SUBSYS	DESCRIPTION	DOC
2SIDES	makes files for multi-columns and/or 2-sided listing	OL

ACCESS	gives a list of subsys's currently available to GUESTs	
ADDMSG	appends a msg to a specified file	
AID	algebraic interpretive dialog conversational lang. HC	
AIFAIL	assembly lang. - early version of FAIL from SU-AI	OL,HC
BAIL	preliminary version of SAIL debugger (on <SAIL>)	OL
BACKUP	short term file loss protection	OL
BANANARD	msg reading program (many extra features)	OL
BASIC	conversational programming lang. (DEC version)	OL,DML,TUG
BCPL	compiler writing and systems programming lang.	HC
BINCOM	binary comparison of files (now replaced by FILCOM)	DAL
BLIS10	compiler for system implementation (DEC version)	OL,HC,TUG
BLIS11	BLISS for the PDP11	
BLISS	compiler for system implementation (TENEXized)	OL,HC(DEC)
BUDGET	budget management program (especially proposals)	OL
BYE	@BYE same as @BREAK (LINKS)	
CALENDAR	calendar management and reminder system	OL,TUG
CAM	the compare and merge program of SOUP see <DOC>SOUP.MANUAL	
CCL	concise command language	OL,DUH
COPYM	reading/writing DECTapes	OL,TUG
CREF	cross-reference assembly listing	OL,DAL
CRSREF	TENEX cross-referencing program (outfile_infile(s))	
DCHANGE	character set conversion for "foreign" tapes	OL
DCHECK	reads blocks of file into core & calls DDT to examine	OL
DDT	debugger (single-stepping added at IMSSS)	OL,TUG,DAL
DED	text-editor (designed for TENEX)	OL
DEOLD	deletes files by cutoff date of last access	OL
DELVER	deletes excess versions of files	TUG
DIABLO	prints final copy of PUB-produced documents on DIABLO	OL
DIRNUM	translates directory name to number for DEC programs	OL
DO	creates or appends a line to a reminder file	OL
DONE	deletes a line from a reminder file	OL
DROP	similar to DELVER, deletes oldest and 2nd newest on *.*	
DTACOP	DECTape to DECTape copy	
DUMPER	reads/writes magnetic tapes	
EE	@EE <program> runs program on your directory as ephemeral	
ES	@ES <program> runs program on <SUBSYS> as ephemeral	
EXTR	"EXTRactor" processes MACRO/FAIL source files to produce .FAI listing of labels defined	
F40	FORTTRAN IV (see also <DOC>FORTTRAN.HELP and <DOC>LISP-FORTTRAN-INTERFACE.HELP )	OL,TUG,DML
FAIL	assembly language (BBN version of FAIL) e also JSYS manual & <DOC>JSYS.INFO)	OL,HC
FED	the final edit program of SOUP see <DOC>SOUP.MANUAL	
FILCHK	checks SAIL programs for loader incompatibilities	OL
FILCOM	complete file comparison package	OL,DAL,TUG
FILDMP	dumps files in variety of formats	OL
FILES	multiple to multiple copies, renames, protections	
FORTRA	FORTTRAN10(version 1A) (see also <DOC>FORTTRAN.HELP)	OL,HC
FREQ	ranks words in text file according to frequency	
FRKCOM	compares an address space with address space of file	TUG
FTP	ARPANET file transfers	TUG
FUDGE2	updates/manipulates files containing rel programs	DAL,TUG
GETDMP	loads into core .dmp file from SU-AI (SAV only to 677777) type filename to * prompt	
GRIPE	sends comments or complaints about system to staff	TUG
HELP	prints out short general help file for SUMEX or help for	



		programs
HOSTAT	prints network site status information	TUG
IFAIL	assembly language (IMSSS version of FAIL)	OL,HC
ILISP	UC Irvine LISP (extension of LISP 1.6)	OL
IMSSS	direct link to IMSSS	
LD	prints SYSTAT-like info including msg if MAIL WAITING	
LINK10	DEC loader	OL,DAL,TUG
LINK11	linker for PDP11 DOS operating system	
LINKSTAT	prints status of IMSSS link	
LISP	INTERLISP-see also <DOC>LISP-FORTRAN-INTERFACE.INFO	OL,HC
LOADER	(from IMSSS)-see <DOC>LINK10-LOADER-DIFFERENCES.HELP	TUG
LOADGT	GT40 standard format loader	
MACRO	assembly lang.-see JSYS manual & <DOC>JSYS.INFO	TUG,DAL
MAILSTAT	info on queued mail	TUG
MANTIS	Fortran debugger	
MLAB	mathematical modeling and graphics package	OL
MULTI	multiple-fork supervisor--switches between forks	
MY-	prints user's valid accountnames	
ACCOUNTS		
NETSTAT	prints info on ARPANET status	TUG
NON	zero-compresses file, options to remove linenumbers, pagemarks, etc.	
PCSAMP	measures the operation of other user programs	TUG
PIP	DEC utilities program	OL,DUH
PIP11	transfers PDP11 DOS DEctapes to/from TENEX files	OL
PNTMAK	converts underlines to suitable format for LPT:	OL
POET	text editor designed for TENEX use	OL
PPL	an interactive extensible programming lang.	TUG
PROFIL	gives freq of execution of SAIL statements	OL,HC
PUB	document preparation lang.	OL
RD	mail reading program (BANANARD is better)	TUG
READMAIL	mail reading program " "	TUG
RECORD	for pseudo-ttys, typescript of job, detaching from running job	OL
REDUCE	symbolic algebraic language	OL
RUNFIL	uses file instead of tty for input commands	TUG
RUNOFF	document-preparation language (DEC not BBN version)	OL
SAIL	ALGOL-like lang.-see also <DOC>LEAP.MANUAL	OL,HC
SCAN	scans multi-directories for a variety of file info	OL
SEARCH	searches multi-text files for English words or SAIL identifiers, can be used with TV editor	OL
SEGSAV	reads .shr & .low files to produce TENEX .sav	OL
SITBOL	compiler version of SNOBOL	OL
SNDMSG	message sender	OL,TUG
SNOBOL	string-processing programming lang.	OL,HC
SORT	stand alone COBOL column-oriented text file sorter	OL,TUG
SOS	text editor	OL
SPELL	spelling checker/corrector for text files (not TENEX)	OL
SRCCOM	compares text files	TUG
SWITCH	switches the format of a reminder file	OL
SYSIN	executes LISP SYSOUT's	OL
TABLE	creates conversion tables for DCHANGE	
TALK	used with LINK command to eliminate need for ;'s	
TAPCNV	reads card image file processed by MTACPY	TUG
TBASIC	TENEXized version of DARTMOUTH BASIC	OL
TCTALK	teleconferencing over ARPANET	OL

TECO	text editor (see TENEX TECO manual)	OL,TUG
TMERGE	merges specified text pages from files into new file	OL
TODAY	lists the contents of today's reminder file	OL
TRITAP	processes magtapes from XEROX, IMSSS, BBN	OL
TTYTRB	used to report terminal line problems	TUG
TTYTST	prints test patterns for diagnosing terminal	TUG
TV	text editor for TEC and DATAMEDIA displays	OL
TVFIX	restores bad TV files (see <DOC>TV.MANUAL)	
TYMSTAT	(for TYMNET lines only) gives measure of current efficiency of TYMNET transmission	
TYPBIN	does an octal dump of a packed file	TUG
TYPREL	analyzes contents of .REL files	TUG
WATCH	continuous on-line monitoring of system activity	TUG
WATCH.IMS	IMSSS version of WATCH	
WHAT	lists the contents of a reminder file	OL
WHO	prints SYSTAT-like information	
WHOIS	looks up username & prints name/address info on user	OL
XED	text-editor (used with BANANARD)	OL
XT	reformats and prints text file	OL
Z	logs jobs off from inferior forks	

## &lt;DOC&gt; DIRECTORY LISTING -

The following is a listing of the <DOC> directory which contains most of the on-line formal documentation about the system and subsystem.

<DOC> 18-MAY-75 16:06:22

2SIDES.HELP;1  
 A-GENERAL.HELP;11  
 A-GUIDE-TO-TENEX-USER'S-GUIDE.INFO;1  
 A-LIST-OF-AVAILABLE-DOCUMENTATION.INFO;7  
 A-SURVEY-OF-THE-DEC-HANDBOOKS.INFO;9  
 ACCOUNT-NAME-USAGE.INFO;1  
 ALL-SUBSYS'S-AVAILABLE-AT-SUMEX.INFO;4  
 BACKUP.HELP;1  
 BAIL.HELP;2  
 BANANARD.MANUAL;2  
 .UPDATE;3  
 BASIC.HLP;1  
 BBN-PROGRAM-VERSION-NUMBER-STANDARDS.INFO;1  
 BLIS10.HLP;2  
 BLISS.HELP;1  
 BSYS.MANUAL;2  
 BUDGET.SAMPLE;2,1  
 .MANUAL;5,4,3  
 CALENDAR.MANUAL;1  
 CCL.HELP;4  
 .UPDATE;1  
 CHECKDSK.HELP;2  
 CHESS.HELP;1  
 CLEAN.HELP;1  
 COPYM.HELP;2  
 CREF.HLP;1  
 .UPDATE;1  
 DCHANG.HLP;1  
 DCHANGE.MANUAL;1  
 DCHECK.HELP;1  
 DDT.SUPPLEMENT;1  
 DEC-HANDBOOK-GLOSSARY-UPDATE.INFO;1  
 DEC/TENEX-COMMAND-EQUIVALENTS.INFO;1  
 DED.MANUAL;1  
 DELOLD.HELP;1  
 DESCRIPTION-OF-SUMEX-AIM-PROJECTS.INFO;3  
 DIABLO.HELP;3  
 DIRNUM.HELP;1  
 DO .HELP;4  
 DUMP.INFO;1  
 EDIT.INFO;1  
 EDITOR-PROGRAM-INTERFACE.INFO;1  
 FAIL.MANUAL;1  
 .HELP;4

FILCHK.HELP;1  
 FILCOM.HLP;4  
 FILDMP.HELP;2  
 FILEX.UPDATE;1  
   .DOC;1  
 FORDDT.DOC;1  
   .HLP;1  
 FORTRA.HLP;1  
 FORTRAN.HELP;2  
 HOW-TO-UPDATE-DOC.INFO;3  
 IDDT.HELP;1  
 ILISP.MANUAL;1  
 INTERROGATE.HELP;4  
 INTRO-TO-SUMEX-AIM-TENEX.INFO;4  
 LEAP.MANUAL;1  
 LINK10.HLP;1  
   .DOC;1  
 LINK10-LOADER-DIFFERENCES.HELP;1  
 LISP.HELP;3  
   .UPDATE;1  
 LISP-FORTRAN-INTERFACE.HELP;2  
 LIST.HELP;3  
 MACRO.HLP;1  
   .DOC;1  
 MINI-DUMP.LISTING;1  
 MLAB.HLP;2  
 NAIVE.PUB;1  
 NSOS.SUPPLEMENT;1  
   .MANUAL;1  
   .INTRO;1  
 OVERVIEW-OF-COMPUTER-SYSTEM.INFO;1  
 PIP.HLP;1  
   .SUPPLEMENT;1  
   .UPDATE;1  
 PIP11.HELP;1  
 PNTMAK.HELP;1  
 POET.HELP;1  
   .MANUAL;1  
 PROFIL.UPDATE;2  
 PROJECTS-AND-ASSOCIATED-USERS.INFO;8  
 PUB.MANUAL;2  
   .HELP;4  
   .UPDATE;9  
 PUB-MANUAL.PUB;10  
 RECORD.MANUAL;1  
 REDUCE.MANUAL;1  
 RUNOFF.MANUAL;2  
 SAIL.HELP;1  
   .SUPPLEMENT;2  
   .UPDATE;3  
   .TENEX-SUPPLEMENT;1  
   .BEGIN-MANUAL;1  
 SAMPLE.PUB;1  
 SCAN.HELP;1  
 SEARCH.MANUAL;3  
   .INFO;3

SEGSAV.HELP;1  
 SETTING-UP-NEW-USER-DIRECTORIES.INFO;1  
 SITBOL.HELP;1  
 SNDMSG.HELP;6  
 SNOBOL.MANUAL;1  
 SORT.HLP;1  
 SOS.UPDATE;6  
     .HELP;4  
     .MANUAL;1  
 SOUP.HLP;1  
     .MANUAL;1  
 SPELL.MANUAL;9  
 SUMEX-JSYS'S.INFO;1  
 SYSIN.HELP;1  
 SYSTEM-SCHEDULE.INFO;3  
 TBASIC.HELP;2  
     .SAMPLE;1  
     .MANUAL;4  
 TCTALK.DOC;1  
 TECO.SAMPLE;1  
     .COMMANDS;1  
     .HELP;1  
     .TEXT1;1  
     .TEXT2;1  
     .SUMMARY;1  
 TELNET.INFO;1  
 TENEX-EXEC-MANUAL-UPDATE.INFO;5  
 TERMINAL-LINKING.INFO;1  
 TMERGE.HELP;3  
 TRITAP.HELP;1  
 TV .UPDATE;8  
     .MANUAL;6  
 TV-STRINGS.PMAP;1  
 TYMNET-INSTRUCTIONS.INFO;1  
 USER-NAME-ADDRESS-PHONE.INFO;29  
 WHOIS.HELP;1  
 XED.HLP;1  
     .MANUAL;1  
 XT .HELP;1  
 142 FILES, 1332 PAGES

## &lt;BULLETINS&gt; DIRECTORY LISTING -

The following is a listing of the <BULLETINS> directory which is a repository of informal or transient information about the system, subsystems, current events, and items of interest.

<BULLETINS> 18-MAY-75 16:06:42

12-MAR-75.SYSLTR;1  
 ARPANET.BBD;1  
 ASCII.BBD;1  
 BASIC.BBD;1  
 BULLETINS.BBD;1  
 CALENDAR.BBD;4,3,1  
 COMPATIBILITY.BBD;1  
 CONSTANTS(PHYSICAL-OR-CHEMICAL).BBD;1  
 DIURNAL-LOADING-WEEKDAYS.MAR;1  
   .3/31/75;1  
   .2/17/75;1  
   .2/24/75;1  
   .3/3/75;1  
   .3/10/75;1  
   .3/17/75;1  
   .3/24/75;1  
   .APR;1  
   .FEB;1  
 DO .BBD;1  
 EDIT.BBD;1  
 EMPLOYMENT-WANTED.BBD;1  
 FILES.BBD;1  
 FORTRAN.BBD;1  
 GOOD-LISP-USAGE.BBD;2  
 GOOD-SYSTEM-USAGE.BBD;1  
 GUEST-LIST.BBD;2  
 IN-WATS.BBD;1  
 KWIC.BBD;1  
 LIBRARY-SAIL.BBD;1  
 LINK10.BBD;1  
 LINKING.BBD;1  
 LISP.BBD;2  
 LIST.BBD;1  
 LOGIN-CMD.BBD;1  
 LOGIN-MESSAGES.BBD;2  
 MACRO.BBD;1  
 MEETINGS.BBD;1  
 MLAB.BBD;1  
 NEW-EXEC.INFO;1  
 OLD-SYSTEM-MESSAGES.BBD;1  
 PDP11-GT40.BBD;1  
 POSITIONS-AVAILABLE.BBD;1  
 PROTECTION.BBD;1  
 RECORD.BBD;1

SAIL.BBD;1  
      ;1  
SEARCH.BBD;1  
SNDMSG-READMAIL.BBD;1  
SORT.BBD;1  
SOS.BBD;1  
SPELL.BBD;1  
SYSTEM-MESSAGES.BBD;1  
TECO.BBD;1  
TENEX.BBD;1  
TESTIMONIALS.BBD;1  
TV .BBD;1  
TV-STRINGS.PMAP;1  
TYMNET.BBD;1  
60 FILES, 149 PAGES

## APPENDIX F

## Networking and Collaborative Research - DENDRAL Project

The following is a preprint of a paper to be presented at the 170th meeting of the American Chemical Society in Chicago during August 1975 - the symposium title is "Computer Networking in Chemistry". This paper will appear in the proceedings and reflects well the orientation and activities of the SUMEX-AIM resource and its collaborating projects (DENDRAL in this case).

NETWORKING AND A COLLABORATIVE RESEARCH COMMUNITY: A  
CASE STUDY USING THE DENDRAL PROGRAMS.

Raymond E. Carhart\*, Suzanne M. Johnson, Dennis H. Smith, Bruce G. Buchanan, R. Geoffrey Dromey, and Joshua Lederberg.

Departments of \*Computer Science, Genetics, and Chemistry, Stanford University, Stanford, California, 94305.

Computer Science is one of the newest, but also one of the least "cumulative" of the sciences. Gordon (1) has recently pointed out the upsetting disparity between the number of potentially sharable programs in existence and the number which are easily accessible to a given researcher. Although some mechanisms exist for the systematic exchange of program resources, for example the World List of Crystallographic Computer Programs (2), a great deal of programming effort is duplicated among different research groups with common interests. The reasons for this are understandable: these groups are separated by geography, by incompatibilities in computer facilities and by a lack of a means to keep abreast of a rapidly changing field.

The emergence of more economical technologies for data communications provides, in principle, a method for lowering these geographical and operational barriers; for creating, through computer networking, remote sites at which functionally specialized capabilities are concentrated. The SUMEX-AIM (Stanford University Medical EXperimental computer - Artificial Intelligence in Medicine) project is an experiment in reducing this principle to practice, in the specific area of artificial intelligence research applied to health sciences.

The SUMEX-AIM computer facility (3) is a National Shared Computing Resource being developed and operated by Stanford University, in partnership with and with financial support from the Biotechnology Resources Branch of the the Division of Research



Resources, National Institutes of Health. It is national in scope in that a major portion of its computing capacity is being made available to authorized research groups throughout the country by means of communications networks.

Aside from demonstrating, on managerial, administrative and technical levels, that such a national computing resource is a viable concept, the primary objective of SUMEX-AIM is the building of a collaborative research community. The aim is to encourage individual participants not only to investigate applications of artificial intelligence in health science, but also to share their programs and discuss their ideas with other researchers. This places a responsibility upon SUMEX-AIM to develop effective means of communication among community members and among the programs they write. It also places responsibility upon those members to design and document programs that readily can be used and understood by others.

Another aspect of the SUMEX facility is providing service to individuals whose interest is in using, rather than developing, the available computer programs. Although this is not a primary consideration, it is an essential part of the growth of these programs. Most of the SUMEX-AIM projects have formed, or are forming, their own user communities which provide valuable "real world" experience. Figure 1 depicts the typical interaction of such a project with its user community, and with other projects. The participation by users in program development is not just restricted to suggestions, but can also include software created by computer-oriented users to satisfy special needs. In some projects, methods are being considered to further promote this kind of participation.

The purposes of this paper are threefold: first, to indicate the range of research projects currently active at SUMEX; second, to describe in detail one of these projects, DENDRAL, which is of particular interest to chemists; and third, to discuss some problems and possible solutions related to networking and community-building.

## I. Research Activities at SUMEX-AIM

The community of participants in SUMEX-AIM can be divided geographically into local (i. e., Stanford-based) projects and remote projects, and below is given a brief description of the major representatives of each. Communication with the remote projects is accomplished through one or both of the communications networks shown in Figure 2. In most cases, connection with SUMEX-AIM from these remote sites involves only a local telephone call to the nearest network "node".

The SUMEX-AIM system is itself undergoing constant improvement which deserves to be called research, and thus a third section is included here to represent system developments.

## Remote projects

The Rutgers project. Originating from Rutgers University are several research efforts designed to introduce advanced methods in computer science - particularly in artificial intelligence and interactive data base systems - into specific areas of biomedical research. One such effort involves the development of computer-based consultation systems for diseases of the eye, specifically the establishment of a national network of collaborators for diagnosis and recommendations for treatment of glaucoma by computer. Another project concerns the BELIEVER program, which represents a theory of how people arrive at an interpretation of the social actions of others. SUMEX-AIM provides an excellent medium for collaboration in the development and testing of this theory. The Rutgers project includes, in addition, several fundamental studies in artificial intelligence and system design, which provide much of the support needed for the development of such complex systems.

The DIALOG project. The DIAGnostic LOGic project, based at the University of Pittsburgh, is a large scale, computerized medical diagnostic system that makes use of the methods and structures of artificial intelligence. Unlike most other computer diagnostic programs, which are oriented to differential diagnosis in a rather limited area, the DIALOG system has been designed to deal with the general problem of diagnosis in internal medicine and currently accesses a medical data base which encompasses approximately fifty percent of the major diseases in internal medicine.

The MISL Project. The Medical Information Systems Laboratory at the University of Illinois (Chicago Circle campus) has been established to explore inferential relationships between analytic data and the natural history of selected eye diseases, both in treated and untreated forms. This project will utilize the SUMEX-AIM resource to build a data base which could then be used as a test bed for the development of clinical decision support algorithms.

Distributed Data-Base System for Chronic Diseases. This project, based at the University of Hawaii, seeks to use the SUMEX-AIM facility to establish a resource sharing project for the development of computer systems for consultation and research, and to make these systems available to clinical facilities from a set of distributed data bases. The radio and satellite links which compose the communication network known as the ALOHANET, in conjunction with the ARPANET, will make these programs available to other Hawaiian islands and to remote areas of the Pacific basin. This project could well have a significantly beneficial effect on the quality of health care delivery in these locations.

Modeling of Higher Mental Functions. A project at the University of California at Los Angeles is using the SUMEX-AIM facility to construct, test, and validate an improved version of the computer simulation of paranoid processes which has been developed. These simulations have clinical implications for the understanding, treatment, and prevention of paranoid disorders. The current interactive version (PARRY) has been running on SUMEX-AIM and has

provided a basis for improvement of the future version's language recognition capability.

### Local Projects

The Protein Crystallography Project. The Protein Crystallography project involves scientists at two different universities (Stanford and the University of California at San Diego), pooling their respective talents in protein crystallography and computer science, and using the SUMEX-AIM facility as the central repository for programs, data and other information of common interest. The general objective of the project is to apply problem solving techniques, which have emerged from artificial intelligence research, to the well known "phase problem" of x-ray crystallography, in order to determine the three-dimensional structures of proteins. The work is intended to be of practical as well as theoretical value to both computer science (particularly artificial intelligence research) and protein crystallography.

The MYCIN project. MYCIN is an evolving computer program that has been developed to assist physician nonspecialists with the selection of therapy for patients with bacterial infections. The project has involved both physicians, with expertise in the clinical pharmacology of bacterial infections, and computer scientists, with interests in artificial intelligence and medical computing. The MYCIN program attempts to model the decision processes of the medical experts. It consists of three closely integrated components: the Consultation System asks questions, makes conclusions, and gives advice; the Explanation System answers questions from the user to justify the program's advice and explain its methods; and the Rule-Acquisition System permits the user to teach the system new decision rules, or to alter pre-existing rules that are judged to be inadequate or incorrect.

The DENDRAL project. This project, being of particular chemical interest, is described in detail in Section II. Through the SUMEX-AIM facility DENDRAL has gained a growing community of production-level users whose experience with the programs is a valuable guide to further development. Although technically users, some members of this community might better be described as collaborators because they have provided SUMEX-AIM with various special-purpose programs which are of interest to other chemists and which extend the usefulness of the DENDRAL programs.

### SUMEX-AIM System Development

Current research activities at SUMEX-AIM are developing along several lines. On a system development level there are ongoing projects designed to make the system more user oriented. Currently, the system can be expected to provide help to the user who is confused about what is expected in response to a certain prompt. A "?" typed by the user, will, in most cases, provide a list of possible responses

from which to choose. Also available in response to typing "HELP" to the monitor is a general help description containing pointers to files likely to be of interest to a new user.

In an effort to facilitate communication between collaborators, a program called CONFER has been developed to provide an orderly method for multiple participant teletype "conference calls". Basically, the program acts as a character processor for all the terminals linked in the conference, accepting input from only one at a time, and passing it out to the remaining terminals. In this way, the conference, in effect, has a "moderator" terminal, thus allowing for a more orderly transfer of ideas and information.

SUMEX-AIM is also aware of the necessity of making its facilities available for trial use by potential users and collaborators. To this end, a GUEST mechanism has been established for persons who wish to have brief, trial access to certain programs they feel may be of value to them, and about which they would like to obtain more knowledge. This provides a convenient mechanism whereby persons, who have been given an appropriate phone number and LOGIN procedure, can dial up SUMEX-AIM and receive actual experience using a program they may only have heard about.

Another area of system development currently being explored at SUMEX-AIM is that of creating a comprehensive "bulletin board" facility where users can file "bulletins", that is, messages of interest to the SUMEX-AIM community. The facility will also alert users to new bulletins which are likely to be of interest to them, as determined by individual user-interest profile.

## II. DENDRAL - CHEMICAL APPLICATIONS OF INTERACTIVE COMPUTING IN A NETWORK ENVIRONMENT

The major research interest of the DENDRAL Project at Stanford University is application of artificial intelligence techniques for chemical inference, focusing in particular on molecular structure elucidation. Portions of our research are in the area of combined gas chromatography/high resolution mass spectrometry and include instrumentation and data acquisition hardware and software development. This area is beyond the scope of this report; we focus instead on the concurrent development of programs to assist chemists in various phases of structure elucidation beyond the point of initial data collection. SUMEX-AIM provides the computer support for development and application of these programs.

Another aspect of our research is our commitment to share developments among a wider community. We feel that several of our programs are advanced enough to be useful to chemists engaged in related work in mass spectrometry and structure elucidation in general. These programs are written primarily in the programming language INTERLISP, and thus are not easily exportable (exceptions are

indicated subsequently). SUMEX-AIM provides a mechanism for allowing others access to the programs without the requirement for any special programming or computer system expertise. The availability of the SUMEX facility over nationwide networks allows remote users to access the programs, in many instances via a local telephone call.

Much of the following discussion is preliminary because our programs have only recently been released for outside use. Some announcement of their availability has been made, and other announcements will occur in the near future, through talks, publications in press, demonstrations and informal discussions. Although most of our experience has been with local users, they have been good models of remote users in that their previous exposure to the actual programs and computer systems is minimal. Their experience has been extremely useful in helping us to smooth out clumsy interactions with programs and to locate and fix program bugs. Such polishing is important for programs which may be utilized by users from widely differing backgrounds with respect to computers, networks and time sharing systems. We are in the processes of building a community of remote users. We actively encourage such use for two reasons: 1) we feel the programs are capable of assisting others in solving certain molecular structure problems, and 2) such experience with outside users will be a tremendous assistance in increasing the power of our programs as the programs are forced to confront new real-world problems.

The remainder of this section outlines the programs which are available via SUMEX, the utilization of these programs in helping to solve structure elucidation problems and the limitations we see to their use. We discuss current applications of the programs to our research and the research of other users to illustrate better the variety of potential applications and to stimulate an interchange of ideas. Where appropriate, we point out current difficulties with the use both of our programs and of SUMEX. New applications and wider use will certainly change the nature of these problems; we strive to solve current problems, but new ones will always arise to take their place.

#### DENDRAL Programs

We have several programs which we employ in dealing with various aspects of problems involving unknown structures. Some of these programs are exportable, while the remainder are available at SUMEX. The availability of each program is discussed below.

Our initial emphasis in studying applications of artificial intelligence for chemical inference was in the area of mass spectrometry(4-6). This emphasis remains because many of our problems require mass spectrometry as the analytical tool of choice in providing structural information on small quantities of sample. More recently, we have been developing a program (CONGEN, below) directed at more general aspects of structure elucidation. This has extended the scope of problems for which we can provide computer assistance.

We will begin, however, with discussion of the mass spectrometry

programs. The examples used in the discussion are characteristic of our current research problems, although we have focused on relatively simple problems to keep the presentation brief. We trace, in what might be chronological terms, the application of the programs to various phases of a structure problem. In this way we hope to illustrate the place of each program in the analysis. We begin by discussing preprocessing of mass spectral data (CLEANUP and MOLION). Subsequent analysis of such data in terms of structure is then covered (PLANNER). The use of CONGEN is discussed for problems which cannot be handled by the previous programs. Finally, we discuss efforts to discover, with the use of the computer, systematics in the behavior of known substances in the mass spectrometer as a means of extending the knowledge of the system for applications in new areas (INTSUM and RULEGEN).

### Programs for Molecular Structure Problems

The first three programs, CLEANUP, the library-search program and MOLION are in a sense utility programs, but all three play a critical role in processing mass spectral data. Subsequent applications of programs (e.g., PLANNER) for more detailed spectral analysis in terms of structure depend on the successful treatment of the data by CLEANUP and MOLION, while the library search program filters out common spectra which need not undergo a full analysis. The examples used are drawn from our collaboration with persons in the Genetics Research Center at Stanford Hospital. The experimental data which are collected are the results of combined gas chromatographic/low resolution mass spectral (GC/LRMS) analysis of organic components (chemically fractionated and derivatized where necessary) of body fluids, e.g. blood, urine. A typical experiment consists of 500-600 individual mass spectra for each fraction, taken sequentially over time as the various components, largely separated from one another, elute from the gas chromatograph and pass into the mass spectrometer. Each mass spectrum consists of the mass analyzed fragment ions of the component(s) in the mass spectrometer at the time the spectrum was taken. Such spectra are related, indirectly, to the molecular structure of the component(s).

CLEANUP(7). The individual mass spectra obtained from fractionated GC/LRMS analysis are quite often poor representations of corresponding spectra taken from pure compounds. They can be contaminated by the presence of additional peaks and/or distortions of the intensities of existing peaks in the spectrum. Fragment ions from either the liquid phase of the GC column or from components incompletely separated by the gas chromatograph are responsible for the contamination. We have developed a program, referred to here as CLEANUP, which examines all mass spectra in a GC/LRMS run, selects those spectra which contain ions other than background impurities, and remove contributions from background and overlapping components. A spectrum results which compares favorably with the spectrum of a pure component. Biller and Biemann (8) have developed a similar but less powerful program.

For example, the CLEANUP program detected components at points marked with a vertical bar in the (partial) plot of total ion current vs. scan number (time), Figure 3. Note that overlapping components were detected under the envelopes of the GC peaks in the region of scans 485-488, 525-529 and 539-552. We focus our attention on the spectrum recorded at scan 492. The raw data, prior to cleanup, are presented in Figure 4 (top). The spectrum resulting from CLEANUP is presented in Figure 4 (bottom). Note that the large ions (e.g., m/e 207, 221 and 315) from background impurities are removed, and that the intensity ratios of peaks at lower masses (e.g., 51 and 77) have been adjusted to reflect their true intensities in the spectrum.

The CLEANUP program is capable of detection of quite low-level components in complex mixtures as indicated by some of the areas of the total ion current plot (Figure 3) where components were detected. It is completely general because nothing in the program code is sensitive to the types of compounds analyzed or the characteristics of possible impurities associated with the compounds or from the GC column. Its major limitation is that mass spectra must be taken repetitively during the course of a GC/MS run. Its performance is enhanced when such spectra are measured closely in time.

The program is offered via SUMEX as an adjunct to use of our other programs; it is not offered as a routine service. Because the program is written in FORTRAN, we routinely use it on our data acquisition computer system so as not to burden SUMEX with tasks better done elsewhere. Similarly, we would assist other frequent users to mount the program on their own systems.

**Library Search.** With a set of "clean" mass spectra available, the next problem is identification of the various components. Over the course of several years, libraries of mass spectral data have been assembled(9). These libraries can be very useful in weeding out from a group of spectra those which represent known compounds(10). Clearly, one should spend time on solving the structures of unknown compounds, not on rediscovering old ones. The CLEANUP program provides mass spectra which are of sufficient quality to expect that known compounds would be identified easily from such libraries.

The spectra detected by CLEANUP in the region of scans 480 to 580 (Figure 3) were matched against the library of biomedically relevant spectra compiled by S. Markey (National Institutes of Health) and our extensions to that library (we wish to thank S. Grotch, Jet Propulsion Laboratory, Pasadena, Ca. for providing some of the library matchings). Excellent matches with the library were obtained for scans 492, 496, 509, 519, 529 and 548. The components are indole acetic acid methyl ester, di-n-butylphthalate, caffeine, salicyluric acid methyl ester, methoxyhippuric acid methyl ester and n-C<sub>24</sub> hydrocarbon respectively (structures given in Figures 3 and 4). Spectra scans at 485, 487, 525, 530, 536, 539, 554 and 576 did not match well with any spectrum in the library and thus must be examined further for structural information. The necessity for preprocessing the data using CLEANUP prior to library matching is illustrated from indole acetic acid methyl ester (scan 492). The "clean" spectrum

(Figure 4, bottom) was matched to the library spectrum of this compound much better than to that of any other compound. The raw spectrum (Figure 4, top), when compared to the library resulted in eleven other compounds which matched more closely than the correct one.

This brief example illustrates the obvious value and limitations of library searching. The most interesting compounds for subsequent analysis are those which are unknown. The fractions of urine extracts are replete with unidentified compounds because of the inadequacy of current library compilations. As new compounds are identified they are, of course, added to the library, so that future analyses need not reinvestigate the same material.

We currently perform library searching on our data acquisition and reduction computer systems. We can, if necessary, offer limited library search facilities via SUMEX. However, because commercial facilities are available (e.g., over the GE network), routine library search service is not available on SUMEX.

MOLION(11). At this stage we are left with a collection of mass spectra of unknown compounds. The library search results may have provided some clues as to the type of compound present, e.g., compound class. Structure elucidation now begins in earnest. The key elements in problems of structure elucidation are the molecular weight and empirical formula of a compound. Without these essential data, the structural possibilities are usually too immense to proceed further. Mass spectrometry is frequently used to determine molecular weights and formulae, but there is no guarantee that the mass spectrum of a compound displays an ion corresponding to the intact molecule. For example, many of the derivatives of the amino acid fractions of urine display no molecular ions. When we are given only the mass spectrum (and for GC/MS analysis a mass spectrum may be all that is available) we must somehow predict likely molecular ion candidates. The program MOLION performs this task. Given a mass spectrum, it predicts and ranks likely molecular ion candidates independent of the presence or absence of an ion in the spectrum corresponding to the intact molecule. The published manuscript(11) provides many examples of the performance of the program.

The mass spectrum of an example, unknown X, (which we will pursue in more detail below) is given in Figure 5. The results obtained from MOLION are summarized in Table I. The observed ion at  $m/e$  263 is ranked as the most likely candidate.



Table I. Results of Molecular Ion Determination for the Unknown Compound, X, whose Mass Spectrum is Presented in Figure 5.

CANDIDATE	RANKING INDEX
263.0	100
307.0	41
299.0	38
295.0	34
281.0	25

The MOLION program is written to operate on either low or high resolution mass spectra. The program has certain limitations which have been summarized in detail previously(11).

MOLION is available on SUMEX. A FORTRAN version, initially for low resolution mass spectra, is being written so that the program can be run on smaller computers and exported to others. However, it will continue to be available via SUMEX so that others can access it easily. MOLION is contained within PLANNER as one of the available methods for detecting candidate molecular ions.

PLANNER(12). The PLANNER program is designed to analyze the mass spectrum of a compound or of a mixture of related compounds. Because there is no ab initio way of relating a mass spectrum of a complex organic molecule to the structure of that molecule, PLANNER requires fragmentation rules for the class of compounds to which the unknown belongs. This is its major limitation. For our example the class was unknown, forcing us to resort to other means of assistance.

Applications and limitations of PLANNER have been discussed extensively(12,13). The program is very powerful in instances where mass spectrometry rules are strong (i.e., general, with few exceptions). In instances where rules are weak or nonexistent, additional work on known structures and spectra may yield useful rules to make PLANNER applicable (see INTSUM and RULEGEN, below). One important feature of PLANNER is its ability to analyze the spectra of mixtures in a systematic and thorough way. Thus, it can be applied to spectra obtained as mixtures when GC/MS data are unavailable or impossible to obtain. PLANNER is available in an interactive version over SUMEX, requiring three kinds of information as input: the high or low resolution mass spectrum, the characteristic skeletal structure for molecules in the specific compound class, and the fragmentation rules for the class. Additional knowledge about the unknown can be used by the program to constrain the structural possibilities.

CONGEN(14,15). Structure problems are usually not solved with mass spectrometry alone. Even when sample size is too limited for obtaining other spectroscopic data, knowledge of chemical isolation and results of derivatization procedures frequently act as powerful

constraints on structural possibilities. Larger amounts of sample permit determination of other spectroscopic data. Taken together, this information allows determination of structural features (substructures) of the molecule and constraints on the plausibility of ways in which the substructures may be assembled. The CONGEN program is capable of providing assistance in solution of such problems.

CONGEN performs the task of construction, or generation, of structural isomers under constraints. The program accepts as input known structural fragments of the molecule ("superatoms") and any remaining atoms (C,N,O,P,...), together with constraints on how they may be assembled. It is based on the exhaustive structure generator(16,17) and extensions(18) which permit a stepwise assembly of structures.

In an interactive session with the program, a user supplies structural information determined by his own analysis of the data (perhaps with the help of the above programs), together with whatever other constraints are available concerning desired and undesired structural features, ring sizes and so forth. The program builds structures in a series of steps, during which a user can interact further with the procedure, for example, to add new constraints. Although very much a developing program, its ability to accept user-inferred constraints from many data sources makes CONGEN a general tool for structure elucidation which we are making available via SUMEX-AIM in its current form.

For the unknown X, the observed fragment ions from the molecular ion (M) at m/e 263 (Figure 5) suggest several structural features when coupled with the knowledge of the chemical derivatization procedures used on this fraction of the urine extract. The ion at m/e 194 represents loss of 69 amu, probably CF<sub>3</sub>, from fragmentation of a trifluoroacetyl derivative of an amine. This suggests the partial structure 2, Figure 5. The ions at m/e 190 (M-74 amu) and m/e 162 (M-101 amu) suggest the characteristic fragmentation of an n-butyl ester resulting from the second derivatization procedure, formation of the n-butyl esters of free carboxylic acid functions. This suggests the partial structure 1, Figure 5. Taken together, all the above information implies (if no other elements are present) that the empirical formula contains an odd number of nitrogen atoms, at least three oxygen atoms, three fluorine atoms and at least seven carbon atoms. Interestingly, there is only one plausible empirical formula under these constraints, C<sub>11</sub>H<sub>12</sub>N<sub>3</sub>O<sub>3</sub>F<sub>3</sub>.

Structural fragments ("superatoms") 1 and 2 were supplied to CONGEN, together with the remaining four carbon atoms and three degrees of unsaturation (that is, rings plus multiple bonds). With no additional constraints, 155 structures result. The inclusion of other plausible constraints (e.g., no allenes, acetylenes, cyclopropenes, cyclobutenes) reduces the number of structural candidates to just the two isomeric forms of 3, Figure 5.

This problem represents a simple example of a large class of such problems. Although a chemist could probably reach the same conclusions quickly in this case, in the general case, piecing together potential solutions is not a trivial task.

Although still a developing program, CONGEN is, capable of considerable assistance in a wide variety of structure problems. Some areas of current application are summarized in the subsequent section. It is already proving its value in structure elucidation problems by suggesting solutions with a guarantee that no plausible alternatives have been overlooked.

The program has a great deal of flexibility. Many of the types of constraints normally brought to bear on structure elucidation problems can be expressed. However, some types of constraints cannot be easily expressed (e.g., disjunctions of features and stereo-constraints). Recent work by our group and Wipke's(19) will make it possible to add considerations of stereoisomerism relatively easily (a good example of collaboration via SUMEX). We are depending on a broad user community to help us guide further development of CONGEN.

#### Programs for Knowledge Acquisition

INTSUM(20) and RULEGEN(21). When the mass spectrometry rules for a given class of compounds are not known, the INTSUM and RULEGEN programs can help a chemist formulate those rules. Essentially, these programs categorize the plausible fragmentations for a class of compounds by looking at the mass spectra of several molecules in the class. All molecules are assumed to belong to one class whose skeletal structure must be specified. Also, the mass spectra and the structures of all the molecules must be given to the program.

INTSUM collects evidence for all possible fragmentations (within user-specified constraints) and summarizes the results. For example, a user may be interested in all fragmentations involving one or two bonds, but not three; aromatic rings may be known to be unfragmented; and the user may be interested only in fragmentations resulting in an ion containing a heteroatom. Under these constraints, the program correlates all peaks in the mass spectra with all possible fragmentations. The summary of results shows the molecules whose spectra display evidence for each particular fragmentation, along with the total (and average) ion current associated with the fragmentation.

The RULEGEN program attempts to explain the regularities found by INTSUM in terms of the underlying structural features around the bonds in question that seem to "direct" the fragmentations. For example, INTSUM will notice significant fragmentation of the two different bonds alpha to the carbonyl group in aliphatic ketones. It is left to RULEGEN to discover that these are both instances of the same fundamental alpha-cleavage process that can be predicted any time a bond is alpha to a carbonyl group.

These programs are part of the so-called Meta-DENDRAL effort, whose general goal is to understand rule formation activities. Both INTSUM and RULEGEN are available as interactive programs on SUMEX, the former being much more highly developed than the latter. Although these programs can be very useful to chemists interested in finding new mass spectrometry rules, they require having the collection of

mass spectra and molecular structure descriptions available in one computer file. Because of this, they have been used mostly by chemists at Stanford.

#### Applications and Resource Sharing

The DENDRAL programs are being developed to serve a broad community of chemists with structure elucidation problems. Our experience is admittedly limited. In this section we discuss some of the applications, both local and from remote sites, where these programs have proven useful.

CLEANUP and MOLION. These programs are in routine use as part of the Genetics Research Center's GC/LRMS efforts. In addition, MOLION has been incorporated as part of PLANNER. Their generality has proven very useful in applications to a variety of GC/MS problems involving structural studies of urinary metabolites.

PLANNER. The planning program has been used to infer plausible placement of substituents around a skeletal structure for numerous test problems in which the class of the sample was known and the fragmentation rules for the class were known. Those tests have resulted in a program that we believe is general. We have applied this program to unknown mixtures of estrogenic steroids(13). We are preparing to use PLANNER for screening mass spectra of marine sterols to identify quickly those spectra of known compounds and to suggest structures for spectra of new compounds.

CONGEN. CONGEN is being used locally and from remote sites in a wide variety of applications. We have used it for construction of ring systems under constraints(22) and for generation of structures of chlorocarbons(23). We have investigated several monoterpenoid and sesquiterpenoid structure problems to suggest solutions and to ensure that all alternatives had been considered. We are currently investigating the scope of terpenoid isomerism. Two problems relating to unknown photochemical reaction products have been analyzed and results used to suggest further experiments. In most cases we do not know the precise problems under study by remote users, only that they are using the program.

CONGEN will perhaps be the most widely used (by remote users) program of those mentioned above as accessible through SUMEX. This is primarily a result of the wider scope of problems which might benefit from use of the program. However, the need for remote users to have their mass spectral data available at SUMEX for analysis present a significant energy barrier to use of the programs which require these data.

INTSUM and RULEGEN. INTSUM is essentially a production program now, and is being used as such in a variety of applications involving correlations of molecular structures with their respective spectra. Recent or current applications include analysis of the mass spectra of progesterones and related steroids, androstanes, macrolide

antibiotics, insect juvenile hormones and phytoecdysones. These studies serve to develop fragmentation rules which, if of sufficient generality, can in turn be used in PLANNER in the study of unknown compounds.

### III. PROBLEMS RELATED TO NETWORKING

During this first year of operation, the SUMEX-AIM facility has encountered a variety of problems arising from its network availability. In most cases, there has been no clear precedent for the handling of these situations, in fact, many problem-areas still reflect the influences of a yet-developing policy. The hope is that this presentation and discussion of problems and their solutions may give foresight to others who contemplate networking or network use. The problems to be discussed can be loosely associated into three classes; those related to the management of the facility, those pertaining to research activities on the system, and those involving psychological barriers to network use.

#### Managerial problems

"Gatekeeping." The most general problem faced by the organizers of the SUMEX-AIM facility is the question of "gatekeeping." In order to insure a high quality of pertinent research, some kind of refereeing system is needed to assess the value of proposed new projects. The organizers of the facility would seem to be the best source of such judgements; yet, because we are both organizers and members of the SUMEX community, there is a danger that our decisions would unfairly favor local priorities. In order to establish credibility in SUMEX-AIM as a truly national resource, a management system has been instituted that allocates a defined fraction (initially 50%) of the SUMEX resource to external users, under the jurisdiction of an independent national committee (the AIM advisory group). The remaining 50%, allocated for local use, contains a portion for flexible experiments outside of local projects, but on our own responsibility.

Choice of computer and operating system. A second management level problem is the choice of a computer and operating system which optimize the usefulness of the facility for a majority of users, and which encourage intercommunication between remote collaborators. Because SUMEX-AIM is intended to be used primarily for applications of artificial intelligence, and because interactive LISP(24) is a primary language in this type of work, the choice of TENEX(25) as an operating system was dictated somewhat by necessity. TENEX incorporates multiple address spaces, thereby allowing multiple "fork" structure and paging, a design which is necessary to create the large-memory virtual machine required by INTERLISP.

The PDP-10 is a popular machine for interactive computing of all

sorts in university research environments, and thus an added benefit of this choice was expected - the possibility of easily transferring to SUMEX programs developed at other sites. Many of these programs were written not under TENEX but under the 10/50 monitor supplied by the manufacturer. Because a large and useful program library was already available under the 10/50 monitor, one of the design criteria of TENEX was compatibility with such programs; when a 10/50 program is run under TENEX, a special "compatibility package" of routines is invoked to translate 10/50 monitor calls into equivalent TENEX monitor calls. Although the concept is sound, we have found that in practice very few programs written for the 10/50 monitor are able to run under TENEX without extensive modification. Other problems with TENEX include weaknesses in the support of peripheral devices and the lack of a default line-editor. The latter has caused a proliferation of editing programs, and some confusion has resulted because editor conventions vary from program to program. These difficulties have dampened somewhat our initial enthusiasm for the TENEX system.

Nonetheless, TENEX provides some features which are crucial to a comfortable network environment. The standard support programs included with this system facilitate both the sending of messages to other users (either at the same site or at other sites on the ARPA network) and the transfer of data and programs from site to site on that network; also, the ability to "link" two or more terminals allows users to communicate easily and immediately. Both the linking and message facility have been found to be invaluable aids in inter-group communications and in such problems as interactive program debugging. When two terminals are linked, their output streams are merged, thus allowing each terminal to display everything typed at the other terminal. Since only the output stream is affected under these circumstances, it is still possible for each terminal to be used to provide input to separate programs, in addition to being used in a conversational mode.

Resource allocation. As noted above, the computational resources of the SUMEX-AIM facility are apportioned by the AIM advisory group and SUMEX management. Some extensions to the basic TENEX system have been made to reflect this apportioning in the actual use of the facility. Basically, it was recognized that users of the facility are members of groups working on specific projects, and it is among these projects that the facility is apportioned. Disk space and cpu cycles are now distributed among groups instead of among individual users. For example, a user may exceed his individual disk allocation somewhat without any ill effect, so long as the total allocation of his group remains within the limits. Similarly, a Reserve Allocation Scheduler has been added to TENEX which tries to match the administrative cycle distribution over a ninety second time frame. Thus a particular group cannot dominate the machine if a lot of its members are logged in at one time.

It is typical for usage of a facility to peak through the middle hours of the day. Indeed, one of the advantages of having users from around the country is the spreading of the load caused by the difference in time zones. Even so, the facility could offer better service if more people would shift their main usage hours toward

either end of the day. To encourage "soft-scheduling" within groups on the system, SUMEX-AIM publishes a weekly plot of diurnal loading . This plot shows the total number of jobs on the system as well as the number of LISP jobs, since these jobs seem to make the biggest demands of system resources. The result has been an increased awareness by users of system loading and a noticeable increase in the number of users at all hours of the night and early morning.

Protection and system security. Protection for a computer system covers a range of ideas. It means the ability to maintain secrecy - for example, to guarantee the privacy of patient records. It also guarantees integrity by assuring that programs and data are not modified by an unauthorized party.

Questions of protection generally become more interesting and complex as more sharing is involved. Consider the example of a proprietary program which generates layouts given a user's circuit data. The program owner demands assurance that he will be paid whenever his program is used and that copies of the program cannot be made. The user wants guarantees that his data sets cannot be destroyed or copied for a competitor. Yet the user must have access to the program and the program must have access to the data. Unable to support such complicated examples of protection, SUMEX-AIM assumes that sharing takes place between friendly users. This is not to imply that issues of protection and sharing have not appeared. For example, in an effort to improve the human engineering of programs for public use, the capability of recording a session has been built into several of the programs. Studied by the program designers to pinpoint confusing aspects of programs, these recordings serve to improve program design. Since the issue of violation of privacy has been raised, some of these programs now request permission to record a session before doing so. At this time, any guarantee of privacy must be provided by the program designer because TENEX itself does not have the ability to render the protection .

The general design for systems offering "state of the art" protection involves a tolerance for failure; that is, if a potential offender succeeds in breaking through some of the defenses, he still does not place the entire computer system at his mercy. Encrypting of data files provides an additional line of defense. This method is used by at least two calendar or appointment programs on the computer. More general purpose facilities to allow users to encrypt and decrypt any of their files whenever they wish are being developed and will be available soon.

Tenex provides the usual keyword protection at login time and a measure of file protection. Owners of a file may assign a protection number which specifies some combination of READ, WRITE, EXECUTE, or APPEND access to a file for owners, members of a group, or other users. This level of protection is basically enough to prevent accidents and most mischief. System programmer's around the country are aware of a number of TENEX bugs which permit this access to be violated. One user of our system found a way to place himself in a mode where he could modify any file on the system. To date, we have no examples of such activity~ actually having a deleterious effect on SUMEX-AIM.

To make the use of SUMEX-AIM programs easily available on a trial basis for prospective users, a "guest" account system has been established. Since this makes logging into SUMEX-AIM so easy, it has invited some misuse by people using those accounts to play the computer games. A proposed extension to the system now being implemented is a special "guest EXEC" which would extend the protection of the TENEX monitor by allowing guest accounts access to only a more restricted set of programs.

File backup. In order to assure the user maximum protection against loss of valuable work, SUMEX operates a multi-level file backup system. In addition to routine file backup system there are facilities to enable the user to selectively archive his or her disk files. By issuing a simple command to the TENEX executive the user can transmit a message to the operator to copy specified files to magnetic tape. Each such file is copied to two magnetic tapes within 24 hours of issuing the archive command. File retrieval is affected by a similar process. The user also has the alternative option of being able to lodge files in a special backup directory. Files are held in this directory until the next exclusive file dump (see below) at which time they are deleted. In this way the user can remove files from his directory at his own choosing knowing they will be archived by the exclusive dump.

On a system level, an effort is made to maintain file backups such that the maximum possible loss, in the event of a crash fatal to the file system, would amount to no more than one day's work. Once each day all files that have been read or written within the last 48 hours are dumped onto magnetic tape. Files that exist for 48 hours are thus held on two separate tapes. The rotation period for files dumped in this way is 60 days. Once each week a full file dump is made to separate disk storage. Each such dump is kept for two weeks at which time it is replaced by a new file dump. Each month there is a full system dump from disk to magnetic tape. Files can be recovered from the system backup by sending a message to the operator specifying the file name(s) and when the file was last read or written (if such information is available).

Excessive demand for production programs. One of the concepts behind the creation of a shared resource is elimination of the problems which arise when large, complex computer programs are exported. Since, in theory, exportability is no longer a problem, there is greater latitude in choice of a language in which program development can take place. In the case of some of the DENDRAL programs, it was thought that program development should take place in INTERLISP, a language that lends itself well to the artificial intelligence nature of these programs, but does not lead to particularly efficient run-time code.

In order to ascertain the usefulness of these programs and to determine what areas remain in need of work, chemist collaborators are being sought. As these users increase in number and begin to use the programs more frequently, it is almost certain that the inherent slowness of the predominately LISP code will affect the whole system as well as handicap the efficient use of the DENDRAL programs.



Additionally, some of the chemist-users who are finding the programs most useful and who are most enthusiastic about their potential use, are persons working in industry. Although, in one sense, this interest from industry could be interpreted as an indication of the "real-world" usefulness of the programs, it came as rather a surprise to both SUMEX and DENDRAL personnel.

The fact that SUMEX-AIM is funded by NIH as a national resource prohibits the facility from providing a service, at taxpayer's expense, to a private industry. Although there is precedent for a site funded via government grant to charge a fee for service, such an arrangement leads to highly complicated bookkeeping, and is contrary to the essential purpose of SUMEX-AIM; to be a research-oriented rather than service-oriented facility. This leaves the industrial users in the position of being more than willing to pay for the use of the programs, but of having no mechanism whereby they can be charged. Furthermore, the fact that the programs are coded in LISP for a highly specialized environment, almost guarantees the impossibility of export, except to an almost identical computer system.

An intermediate solution that will help to solve the problem of industrial users on SUMEX and will help to alleviate the system loading resulting from heavy usage of LISP coded production programs, is to mount CONGEN on a closely related computer which is operated on a fee for service basis. However, in order to make this program available at a reasonable fee, it has become evident that it will be necessary to recode the LISP sections of the program into a more efficient and easily exportable language.

#### Research-oriented problems

Community mindedness. Those involved in computer science research at SUMEX face a general problem which is absent or greatly lessened at non-network sites; the problem of community mindedness. The network provides a large and varied set of other researchers and users who have an interest in their work. Although the network-TENEX combination provides new forms of communication with these remote parties, the traditional means of fully describing the use and structure of a complex program, a detailed person-to-person discussion, is not convenient. Comprehensive documentation gains importance in such a situation, and within the DENDRAL project a great deal of time has been needed in the development of program descriptions which are adequate for a diverse audience. Also, in both DENDRAL and MYCIN, effort has been and is being directed toward "human engineering" in program design; to provide the user with commands which assist him in using the programs, in understanding the logic by which the programs reach certain decisions and in communicating questions or comments on the programs' operation to those responsible for development. Such "housekeeping" tasks can often be neglected, yet are quite important in smoothing interaction with the community.

Choice of programming language. High level programming languages which are designed for ease of program development are

frequently poor as production-level languages. This is because developmental languages free the researcher from a raft of programming details, thus allowing him to concentrate upon the central logical issues of the problem, but the automatic handling of these details is seldom optimal. Also, because such languages tend to be specialized for certain computers and operating systems, the exportation of programs can be a serious problem. One solution to these problems is the recoding of research-level programs into more efficient language when fast and exportable versions are needed.

Networking greatly eases the problem of exportability, but can also aggravate the the problem of efficiency. As mentioned in the previous section, the DENDRAL programs, which are undergoing constant development, found a substantial number of production-level users. Because of the inefficiencies of INTERLISP (a 50- to 100-fold improvement in running time is not uncommon when an INTERLISP program is translated into FORTRAN), this use adversely impacted the entire system. Because the DENDRAL programs are quite large and complex, their translation into other languages is impractically tedious. A partial solution to this problem is provided by the TENEX operating system, which allows some interface between programs written in different languages. With such intercommunication, time-consuming segments of an INTERLISP program which are not undergoing active development can be reprogrammed in another more efficient language. The developmental parts of the program are left in INTERLISP, where modifications can easily be made and tested. The CONGEN program uses three languages; INTERLISP, FORTRAN and SAIL(26). The SAIL segment was added when a new feature, whose implementation was fairly straightforward, was included in CONGEN. Since then, the SAIL portion gradually has been taking over some of the more time-consuming tasks. This method allows a balance in the trade-off between ease of program development and efficiency of the final program.

Accumulation of expert knowledge in knowledge-based programs. Just as statistics-based programs need to worry about accumulation of large data bases, knowledge based programs need to worry about the accumulation of large amounts of expertise. The performance of these programs is tied directly to the amount of knowledge they have about the task domain -- in a phrase, knowledge is power. Therefore, one of the goals of artificial intelligence research is to build systems that not only perform as well as an expert but that also can accumulate knowledge from several experts.

Simple accretion of knowledge is possible only when the "facts", or inference rules, that are being added to the program are entirely separate from one another. It is unreasonable to expect a body of knowledge to be so well organized that the facts or rules do not overlap. (If it were so well organized, it is unlikely that an artificial intelligence program would be the best encoding of the problem solver.) One way of dealing with the overlap is to examine the new rules on an individual basis, as they are added to the system in order to remove the overlap. This was the strategy for developing the early DENDRAL programs. However, it is very inefficient and becomes increasingly more difficult as the body of knowledge grows.

The problem of removing conflicts, or potential conflicts, from overlapping rules becomes more acute when more than one expert adds new rules to the knowledge base. Of course, the advantages of allowing several experts to "teach" the system are enormous -- not only is the program's breadth of knowledge potentially greater than that of a single expert, but the rules are more apt to be refined when looked at by several experts. On the other hand, one can expect not only a greater volume of new rules but a higher percentage of conflicts when several experts are adding rules.

Having a computer program that can accumulate knowledge presupposes having an organization of the program and its knowledge base that allows accumulation. If the knowledge is built into the program as sequences of low-level program statements -- as often happens -- then changing the program becomes impossible. Thus current artificial intelligence research stresses the importance of separating problem-solving knowledge from the control structure of the program that uses that knowledge.

Another problem, at a political rather than a programming level, becomes apparent with one accumulation process: how does the program distinguish an expert from a novice? In the MYCIN program we have circumvented the problem by having the program ask the current user for a keyword that would identify him as an expert. It is then a bureaucratic decision as to which users are given that keyword. There is nothing subtle in this solution, and one can imagine far better schemes for accomplishing the same thing. The point here is that not every user should have the privilege of changing rules that experts have added to the system, and that some safeguards must be implemented.

#### "Human nature" barriers to SUMEX use

Countering disbelief. There is sometimes a tendency among those unfamiliar with the capabilities and limitations of computers and computer programs to express disbelief. This is not disbelief in the sense of worrying that the programs have errors and produce erroneous results. Indeed, the fact that a problem is being done by a computer seems to generate some faith that it might be right, or at least significantly reduces questions about correctness. The disbelief is that programs, which are designed to model, or to emulate, human problem solving will not be capable of useful performance. This, of course, is the classic argument against artificial intelligence -- we think in mysterious ways and have such a complex brain that a computer program must be inferior. In some cases, authors of artificial intelligence programs have brought such criticism upon themselves by not stressing limitations, or by making extravagant claims.

In the DENDRAL project, we have tried to counter this type of disbelief in a number of ways. We have tried to stress that our programs are designed to assist, not replace chemists. We have always discussed limitations to give a reasonable perspective on capabilities vs. limitations of a program. Most importantly however, we have

focused on those aspects of problems which are amenable to systematic analysis, i.e., those problems which can be done manually, but only with difficulty and with the consumption of a great deal of time which a chemist could better spend on more productive pursuits. Examples of this would include the application of PLANNER to mixtures where all fragmentations may have to be considered as possible fragments of every molecular ion, the systematic analysis by INTSUM of possible fragmentation processes, the consideration by MOLION of all plausible possibilities, and the structure generation capabilities of CONGEN.

We have also tried to reduce chemists' disbelief by blurring the "outsider-insider" distinction, in particular by having trained chemists work on the programs and make them useful to themselves first. Further, when "outside" chemists are first introduced to the programs, the introduction is done by another chemist who has already thought through and can readily explain many of the chemistry-related problems.

The ultimate way to counter disbelief, however, is to illustrate high levels of performance. If a potential user is aware of the goals (intent) of a program and its limitations, a few examples of results which would be extremely difficult to obtain without the program are very convincing.

The "security" of a local facility. Networking is still a relatively new concept to many people, and there is a resistance to departing from the "traditional" modes of computing. There is a sense of security in having a local computing facility with knowledgeable consultants within walking distance, and in having "hard" forms of input (eg, boxes of computer cards) and output (eg, voluminous listings). These props are difficult to simulate over a network connection - in most cases a user's interaction with the remote site takes place exclusively through a computer terminal - yet the quality of service can match or exceed that of a local facility; programs and large data sets can be entered and stored on secondary storage as can large output files; all types of program and data editing can be done with interactive editing programs; programs can be written in an interactive mode so that small amounts of control information can be input and key results output in "real time" over the terminal; And as noted in a previous section, consultation can be significantly more productive providing that the remote operating system supports the appropriate types of communication possibilities.

There can, of course, be no denying that there are problems in learning to use a distant computer system, be it for program development or for the use of certain programs. Whether or not overcoming these problems to gain access to the special resources which are available, is worth the effort, is a question answerable only by the individuals involved. Fortunately, there will always be those persons who have a pressing problem in need of solution and who are willing to try a new approach; regardless of whether or not they have had prior network experience.

#### IV. THE SUMEX-AIM FACILITY

The SUMEX-AIM computer facility consists of a Digital Equipment Corporation model KI-10 central processor operating under the TENEX time sharing monitor. It is located at Stanford University Medical Center, Stanford, California.

The system has 256K words (36 bit) of high speed memory; 1.6 million words of swapping storage; 70 million words of disk storage; two 9-track, 800 bpi industry compatible tape units; one dual DEC-tape unit; a line printer; and communications network interfaces providing user terminal access via both TYMNET and ARPANET.

Software support has evolved, and will continue to evolve, based on user research goals and requirements. Major user languages currently include INTERLISP, SAIL, FORTRAN-10, BLISS-10, BASIC and MACRO-10. Major software packages available include OMNIGRAPH, for graphics support of multiple terminal types, and MLAB, for mathematical modeling.

The SUMEX-AIM computer generally is left with no operator in attendance; thereby helping to eliminate some overhead, but also creating some problems. Users who wish to run jobs requiring tapes must make arrangements to mount their own tapes. Likewise, obtaining listings from the line printer can be somewhat difficult since there is no regular schedule for distribution of this output. The solution to these two problems has been to make keys to the machine room available at strategic locations, convenient to all groups of local users. This experiment in basic "resource sharing" has not resulted in any of the major problems one might expect from having a fairly large group of people with hands-on access to a computer.

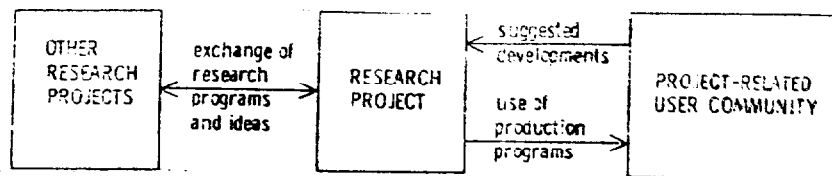


Figure 1. Interactions in the SUMEX-AIM Community

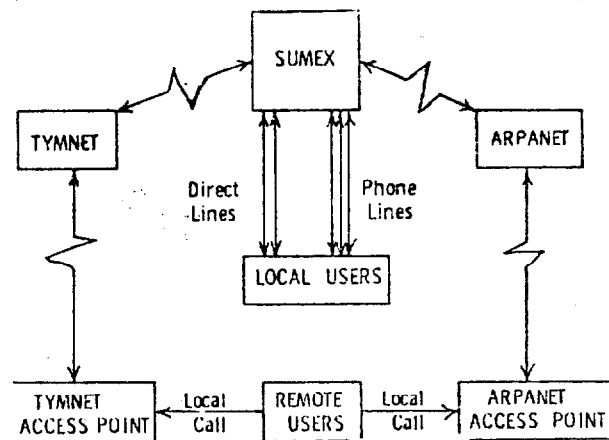


Figure 2. Access to SUMEX-AIM

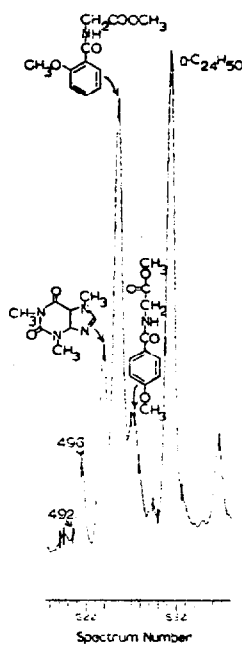


Figure 3. Total Ion Current vs. Spectrum Number in a GC/LRMS Run

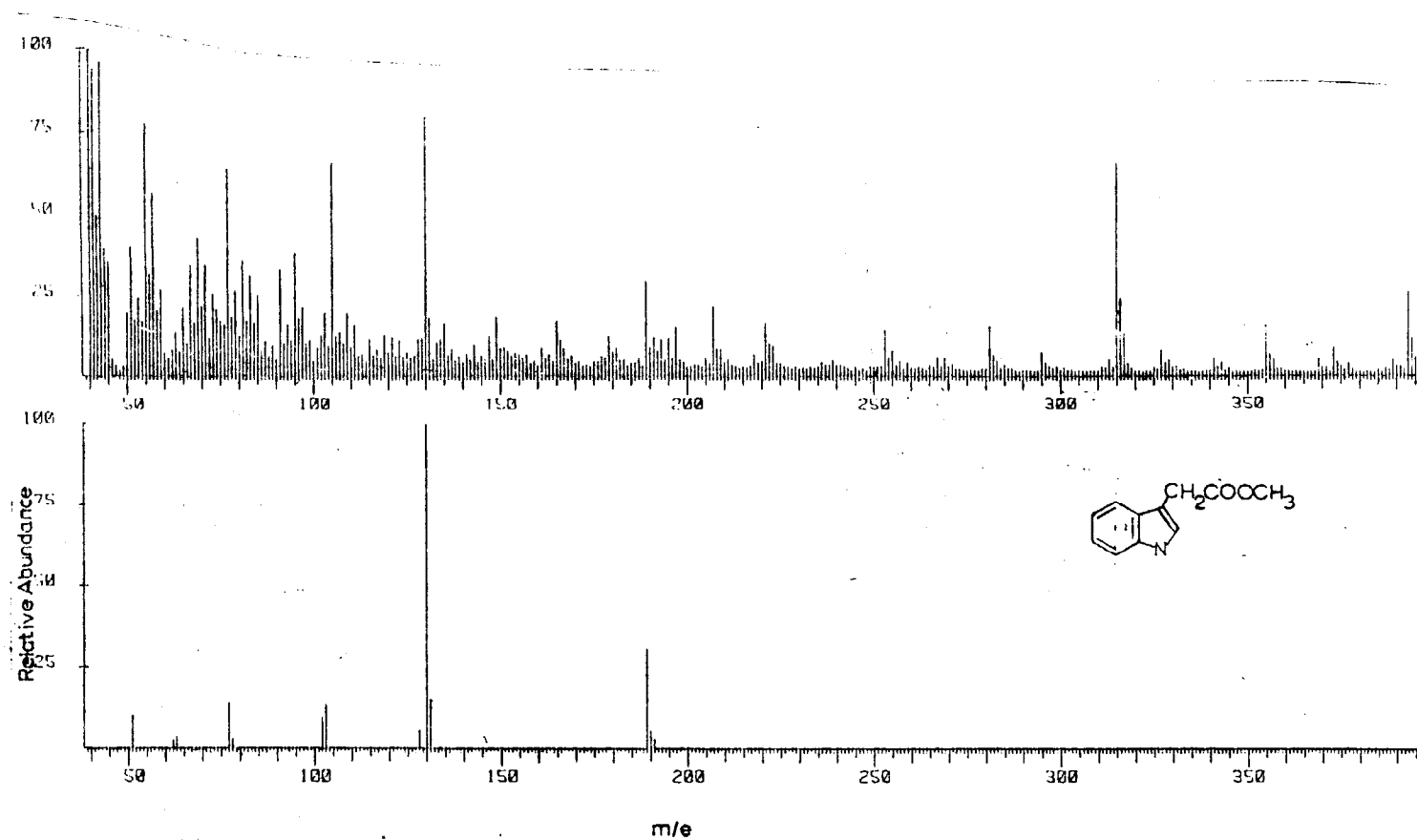


Figure 4. Spectrum number 492 from the GC/LRMS trace shown in Figure 3:  
(top) Raw data; (bottom) Spectrum output by CLEANUP

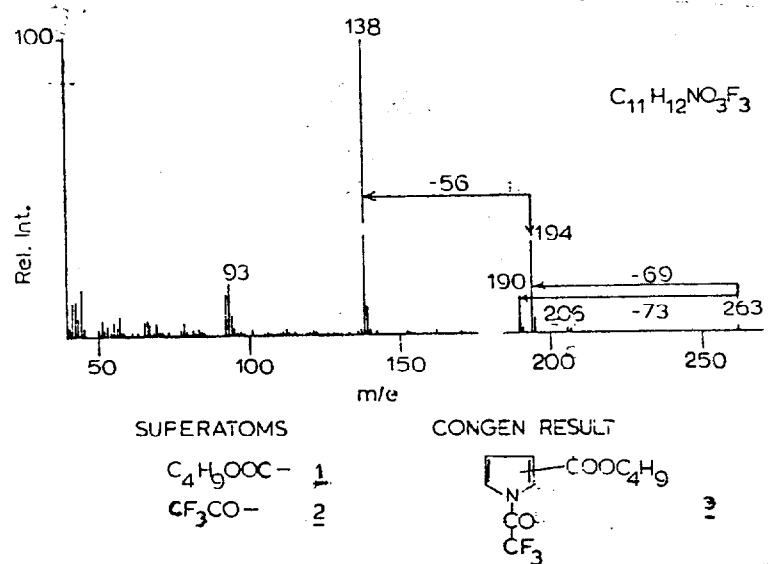


Figure 5. Low-resolution Mass Spectrum of Unknown X. The indicated superatoms were deduced from the spectrum and the chemical history of the sample. Based on these and other constraints, CONGEN obtains the indicated result.



## REFERENCES

- (1) Gordon, R. M., *Datamation*(1975), 21(2), 127.
- (2) "World List of Crystallographic Computer Programs," Second Edition, D. P. Shoemaker, Bronders-Offset, Rotterdam, 1966.
- (3) Professor Joshua Lederberg, Principle Investigator.
- (4) Lederberg, J., Sutherland, G. L., Buchanan, B. G., Feigenbaum, E. A., Robertson, A. V., Duffield, A. M. and Djerassi, C., *J. Amer. Chem. Soc.*(1969), 91, 2973.
- (5) Duffield, A. M., Robertson, A. V., Djerassi, C., Buchanan, B. G., Sutherland, G. L., Feigenbaum, E. A. and Lederberg, J., *J. Amer. Chem. Soc.*(1969), 91, 2977.
- (6) Buchanan, B. G., Duffield, A. M. and Robertson, A. V., "Mass Spectrometry: Techniques and Applications," G. W. A. Milne, Ed., p. 121, John Wiley and Sons, New York, 1971.
- (7) Dromey, R. G., unpublished results, preprint available on request, Dept. of Computer Science, Serra House, Stanford University, Stanford, Calif. 94305.
- (8) Biller, J. E. and Biemann, K., *Anal. Lett.*(1974), 1974, 515.
- (9) Several libraries of mass spectral data are available in various forms. The Aldermaston Data Centre (see the "Mass Spectrometry Bulletin") can provide information on the availability of such libraries.
- (10) Hertz, H. S., Hites, R. A. and Biemann, K., *Anal. Chem.*(1971), 43, 681.
- (11) Dromey, R. G., Buchanan, B. G., Smith, D. H., Lederberg, J. and Djerassi, C., *J. Org. Chem.*(1975), 40, 770.
- (12) Smith, D. H., Buchanan, B. G., Engelmores, R. S., Duffield, A. M., Yeo, A., Feigenbaum, E. A., Lederberg, J. and Djerassi, C., *J. Amer. Chem. Soc.*(1972), 94, 5962.
- (13) Smith, D. H., Buchanan, B. G., Engelmores, R. S., Adlerkreutz, H. and Djerassi, C., *J. Amer. Chem. Soc.*(1973), 95, 6078.
- (14) Smith, D. H. and Carhart, R. E., Abstracts, 169th Meeting of the American Chemical Society, Philadelphia, April 6-11, 1975.
- (15) Carhart, R. E., Smith, D. H., Brown, H. and Djerassi, C., *J. Amer. Chem. Soc.*, submitted for publication.
- (16) Masinter, L. M., Sridharan, N. S., Lederberg, J. and Smith, D. H., *J. Amer. Chem. Soc.*(1974), 96, 7702.

- (17) Masinter, L. M., Sridharan, N. S., Carhart, R. E. and Smith, D. H., J. Amer. Chem. Soc.(1974), 96, 7714.
- (18) Brown, H., SIAM Journal of Computing, submitted for publication.
- (19) Wipke, W. T. and Dyott, T. M., J. Amer. Chem. Soc.(1974), 96, 4825.
- (20) Smith, D. H., Buchanan, B. G., White, W. C., Feigenbaum, E. A., Djerassi, C. and Lederberg, J., Tetrahedron(1973), 29, 3117.
- (21) Buchanan, B. G., to appear in the Proceedings of NATO Advanced Study Institute on Computer Oriented Learning Processes, 1974, Bonas, France.
- (22) Carhart, R. E., Smith, D. H. and Brown, H., J. Chem. Inf. Comp. Sci., in press (May, 1975).
- (23) Smith, D. H., Anal. Chem., in press (May 1975).
- (24) Teitelman, W., "INTERLISP Reference Manual," Xerox Corp. (Palo Alto Research Center), Palo Alto, Calif., 1974.
- (25) Bobrow, D. G., Burchfiel, J. D. and Tomlinson, R. S., Commun. ACM(1972), 15(3), 135.
- (26) VanLehn, K. A., "SAIL User Manual," Stanford Artificial Intelligence Laboratory, Stanford, Calif., 1973.

## APPENDIX G

## Management Committee Membership

The following are the membership lists of the various SUMEX-AIM management committees at the present time:

## AIM EXECUTIVE COMMITTEE:

=====

LEDERBERG, Dr. Joshua (LEDERBERG) (Chairman)  
Department of Genetics, S331  
Stanford University Medical Center  
Stanford, California 94305  
(415) 497-5801

AMAREL, Dr. Saul (AMAREL)  
Department of Computer Science  
Rutgers University  
New Brunswick, New Jersey 08903  
(201) 932-3546

BREWER, Dr. Carl R. (BREWER)  
Biotechnology Resources Branch  
National Institutes of Health  
Building 31, Room 5B25  
9000 Rockville Pike  
Bethesda, Maryland 20014  
(301) 496-5411

LINDBERG, Dr. Donald (LINDBERG) (Adv Grp Member)  
605 Lewis Hall  
University of Missouri  
Columbia, Missouri 65201  
(314) 882-6966

## AIM ADVISORY GROUP:

=====

LINDBERG, Dr. Donald (LINDBERG) (Chairman)  
 605 Lewis Hall  
 University of Missouri  
 Columbia, Missouri 65201  
 (314) 882-6966

AMAREL, Dr. Saul (AMAREL)  
 Department of Computer Science  
 Rutgers University  
 New Brunswick, New Jersey 08903  
 (201) 932-3546

BREWER, Dr. Carl R. (BREWER) (Executive Secretary)  
 Biotechnology Resources Branch  
 National Institutes of Health  
 Building 31, Room 5B25  
 9000 Rockville Pike  
 Bethesda, Maryland 20014  
 (301) 496-5411

BOBROW, Dr. Daniel G. (BOBROW)  
 Xerox Palo Alto Research Center  
 3333 Coyote Hill Road  
 Palo Alto, California 94304  
 (415) 494-4438

FEIGENBAUM, Dr. Edward (FEIGENBAUM)  
 Serra House  
 Department of Computer Science  
 Stanford University  
 Stanford, California 94305  
 (415) 497-4878

FELDMAN, Dr. Jerome (FELDMAN)  
 Department of Computer Science  
 University of Rochester  
 Rochester, New York  
 (716) 275-5478

LEDERBERG, Dr. Joshua (LEDERBERG) (Ex-officio)  
 Principal Investigator - SUMEX  
 Department of Genetics, S331  
 Stanford University Medical Center  
 Stanford, California 94305  
 (415) 497-5801

MILLER, Dr. George (GMILLER)  
 The Rockefeller University  
 1230 York Avenue  
 New York, New York 10021  
 (212) 360-1801

REDDY, Dr. D.R. (REDDY)  
 Department of Computer Science

Carnegie-Mellon University  
Pittsburgh, Pennsylvania  
(412) 621-2600, Ext. 149

SAFIR, Dr. Aran (SAFIR)  
Department of Ophthalmology  
Mount Sinai School of Medicine  
City University of New York  
Fifth Avenue and 100th Street  
New York, New York 10029  
(212) 369-4721

## STANFORD COMMUNITY ADVISORY COMMITTEE

=====

LEDERBERG, Dr. Joshua (LEDERBERG) (Chairman)

Department of Genetics, S331  
Stanford University Medical Center  
Stanford, California 94305  
(415) 497-5801

FEIGENBAUM, Dr. Edward (FEIGENBAUM)

Serra House  
Department of Computer Science  
Stanford University  
Stanford, California 94305  
(415) 497-4878

GREENES, Robert A., M.D.

Department of Community and  
Preventative Medicine, A152  
Stanford University Medical Center  
Stanford, California 94305  
(415) 497-5492

LEVINTHAL, Dr. Elliott C. (LEVINTHAL)

Department of Genetics, S047  
Stanford University Medical Center  
Stanford, California 94305  
(415) 497-5813

## APPENDIX H

## User Information - General Brochure

## SUMEX-AIM

Revised May 1975

The Stanford University Medical Experimental Computer (SUMEX) was established in January, 1974, to provide the first shared national computing facility for medical research. Directed by Dr. Joshua Lederberg, Professor and Chairman of the Department of Genetics, SUMEX is an innovative effort to help biomedical scientists meet today's research requirements and to explore computer applications in many health fields ranging from basic research to bedside care. The project is funded by a grant from the Division of Research Resources of the National Institutes of Health (Biotechnology Resources Branch) for an initial term that expires in July 1978.

At present, SUMEX consists of a powerful PDP-10 computer available to approved users throughout the United States over a computer communication network on a time-shared basis. The project's goals over the next 5 years are: 1) the encouragement of applications of artificial intelligence in medicine (AIM), and 2) the managerial, administrative and technical demonstration of a nationally-shared technological resource for health research.

Such a resource offers scientists both a significant economic advantage in sharing expensive equipment and a greater opportunity to share ideas about their research. This is especially true in computer science, a field whose intellectual and technological complexity has made it difficult to avert the development of relatively isolated research groups. Each group may then tend to pursue its line of investigation with limited convergence on working programs available from others. The SUMEX-AIM project seeks to lower these barriers to scientific cooperation in the field of artificial intelligence applied to health research.

## ARTIFICIAL INTELLIGENCE

The term "artificial intelligence" (AI) refers to research efforts aimed at studying and mechanizing information processing tasks that generally have been considered to require human intelligence. The current emphasis in the field is to understand the underlying principles in efficient acquisition and utilization of material knowledge and representation of conceptual abstractions in reasoning, deductive, and problem-solving activities. AI systems are characterized by complex computational processes that are primarily

non-numeric, e.g., graph-searching and symbolic pattern analysis. They involve procedures whose execution is controlled by different types and forms of knowledge about a given task domain, such as models, fragments of "advice", and systems of constraints or heuristic rules. Unlike conventional algorithms commonly based on a well-tailored method for a given task, AI procedures typically use a multiplicity of methods in a highly conditional manner--depending on the specific data in the task and a variety of sources of relevant information. The tangible objective of this approach is the production of computer programs which, using formal and informal knowledge together with mechanized hypothesis formation and problem-solving procedures, will be more general and effective consultative tools for the clinician and medical scientist.

Each authorized project in the SUMEX-AIM community is concerned in some way with the application of these principles to medical problems. This type of "intelligent" assistance by computer program is perhaps best illustrated by the following brief descriptions of some SUMEX-AIM projects.

#### DENDRAL

The DENDRAL project at Stanford, under the direction of Dr. Lederberg, Professor Edward Feigenbaum, Computer Science, and Professor Carl Djerassi, Chemistry, is aimed at assisting the biochemist in interpreting molecular structures from mass spectral and other chemical information. In cases where the characteristic spectrum of a compound is not catalogued in a library, the DENDRAL programs carry out the rather laborious processes a chemist must go through to interpret the spectrum from "first principles". By symbolically generating "reasonable" candidate structures from hints within the spectrum and a knowledge of organic chemistry and mass spectrometry, the program infers the unknown structure to be the one which best explains the observed spectrum. There is no direct algorithmic path available to determine such a molecular structure from the spectral data--only the inferential process of hypothesis generation and testing within the domain of reasonable solutions defined by a knowledge of organic and physical chemistry.

This process, as implemented in the computer, is a simplified example of the cycle of inductive hypothesis--deductive verification that is often taught as a model of the scientific method (Whether this is a faithful description of contemporary science is arguable, and how it may be implemented in the human brain is unknown. Regardless, these are useful leads rather than absolute preconditions for the pragmatic improvement of mechanized intelligence for more efficient problem-solving.). The elaboration of these approaches with existing hardware and software technologies is the most promising approach to enhancing computer application to the vaguely structured problems that dominate our task domains.

#### THE RUTGERS PROJECT COMPUTERS IN BIOMEDICINE



Professor Saul Amarel, a Rutgers University computer scientist, directs several research efforts designed to introduce advanced methods in computer science--particularly in artificial intelligence and interactive data-base systems--into specific areas of biomedical research.

For example, a group of computer scientists led by Professor Casimir Kulikowski is developing computer-based consultation systems for diseases of the eye in collaboration with Dr. Aran Safir, an ophthalmologist from the Mount Sinai School of Medicine. An important development in this area is the establishment of a national network of collaborators for computer diagnosis and treatment of glaucoma. The computer system, which includes an elaborate pathophysiologic model of the disease, is being tested through the SUMEX-AIM network at three eye centers: Mount Sinai Hospital and Medical Center, New York; Washington University, St. Louis; and The Johns Hopkins University, Baltimore. Glaucoma, in one form or another, affects 2% of all people over 40 years of age. It is a disease in which increased pressure within the eye may lead to irreparable optic nerve damage and blindness. The computer-based program has great potential for assisting clinicians and researchers in understanding the disease, diagnosing it more accurately and improving its treatment.

In another project, Professor Charles Schmidt, a social psychologist, is developing a theory of how people arrive at interpretation of the social actions of others. The theory will be tested in situations such as the psychiatric interview and the legal trial. The computer system which currently represents the theory is called "Believer". It includes a large body of statements about people's motivations and actions. The SUMEX-AIM environment provides an excellent medium for collaboration between Dr. Schmidt's group and other researchers around the Country in the development and testing of this computer-based theory.

The Rutgers project includes, in addition, several fundamental studies in artificial intelligence and system design. These provide much of the support needed for the development of complex systems such as the glaucoma consultation and the "Believer" programs.

#### MYCIN Computer-based Consultation in Clinical Therapeutics

Dr. Stanley Cohen, Associate Professor and Head of the Division of Clinical Pharmacology at Stanford, directs this research in collaboration with Dr. Stanton Axline and with computer scientists interested in artificial intelligence and medical computing. An evolving computer program developed to assist physician nonspecialists in the selection of therapy for patients with bacterial infections, MYCIN attempts to model the decision processes of medical experts. It consists of three closely integrated components: the Consultation System asks questions, makes conclusions and gives advice; the Explanation System answers questions from the user to justify the program's advice and explain its methods; and the Rule-Acquisition

System permits the user to teach the system new decision rules or to alter pre-existing rules judged to be inadequate or incorrect. Goals for further development of the system include expansion of the consultation program to deal with infections other than bacteremias and implementation and evaluation of the system in the clinical setting at Stanford University Hospital.

#### COMPUTING APPLIED TO PROTEIN CRYSTALLOGRAPHY

Members of the artificial intelligence project at Stanford also are collaborating with Professor Joseph Kraut and Dr. Stephan Freer, protein crystallographers at the University of California, San Diego. They are using the SUMEX-AIM facility as the central repository for programs, data and other information of common interest. The general objective of the project is to apply problem-solving techniques, which have emerged from artificial intelligence research, to the well-known "phase problem" of x-ray crystallography in order to determine the three-dimensional structures of proteins. The work is intended to be of both practical and theoretical value to computer science (particularly artificial intelligence research) and protein crystallography.

#### DIALOG

The DIAGnostic LOGic project, under the direction of Dr. Harry Pople and Dr. Jack Myers at the University of Pittsburgh, is a large-scale, computerized medical diagnostic system utilizing the methods and structures of artificial intelligence. Unlike most computer diagnostic programs, which are oriented to differential diagnosis in a rather limited area, the DIALOG system deals with the general problem of diagnosis in internal medicine and currently accesses a medical data base encompassing approximately 50% of the major diseases in internal medicine.

#### MISL

The Medical Information Systems Laboratory at the University of Illinois at Chicago Circle has been established under the direction of Dr. Bruce McCormick, Information Engineering, in collaboration with Dr. Morton Goldberg, an ophthalmologist at the U of I medical school. The project explores inferential relationships between analytic data and the natural history of selected eye diseases both in treated and untreated forms. SUMEX-AIM will be utilized to build a data base to be used as a test bed for the development of clinical decision support algorithms.

#### DISTRIBUTED DATA-BASE SYSTEM FOR CHRONIC DISEASE

This project, based at the University of Hawaii, is under the

direction of Dr. Franklin Kuo of the Department of Electrical Engineering and Technical Director of the ALOHA System. It seeks to use SUMEX-AIM to establish a resource-sharing project for development of computer systems for consultation and research and to make these systems available to clinical facilities from a set of distributed data bases. Radio and satellite links composing the ALOHANET communication network, in conjunction with the ARPANET, will make these programs available to other Hawaiian islands and to remote areas of the Pacific basin. This project could have a significantly beneficial effect on the quality of health care delivery in these locations.

#### SUMEX-AIM Management

A significant part of the SUMEX-AIM experiment is the development of a management structure to maximize the utility of the computer capability for a national community.

Users of the SUMEX facility are divided for administrative purposes into two groups: 1) those at Stanford University School of Medicine, and 2) those elsewhere in the United States. The facility resources (computing capacity and consulting support) are allocated in equal portions to the two groups. As Principal Investigator for the SUMEX grant, Dr. Lederberg reviews Stanford medical school projects with the assistance of a local advisory committee. National users may gain access to the facility resources through an advisory panel for a national program in artificial intelligence in medicine (AIM). The AIM Advisory Group consists of members-at-large of the AI and medical communities, facility users and the Principal Investigator of SUMEX as an ex-officio member. A representative of the National Institutes of Health-Biotechnology Resources Branch (NIH-BRB) serves as Executive Secretary.

The SUMEX-AIM computing resource is initially allocated to qualified users without fee. This, of course, entails a careful review of the merits and priorities of proposed applications. At the direction of the Advisory Group, expenses related to communications and transportation to allow specific users to visit the facility may be covered as well.

SUMEX-AIM is aware of the necessity of making the facility available for trial use by potential users and collaborators. A GUEST mechanism has been established for those who have an indicated requirement for brief access to certain programs. Those who have been given an appropriate telephone number and login procedure can dial up SUMEX-AIM to exercise these programs on a trial basis.

#### USER QUALIFICATIONS

Applications for use of the SUMEX-AIM facility are judged on the basis of:

- 1) The scientific interest and merit of the proposed research.
- 2) The relevance of the research to the artificial intelligence approach of SUMEX-AIM as opposed to other computing alternatives.
- 3) The user's prospective contributions and role in the community, e.g., developing and sharing new systems or applications programs, sharing use of special hardware, etc.
- 4) The user's capability and intentions of operating in a community-effective style for mutual advantage. Besides the programming innovations that some participants may contribute, all are expected to furnish expert knowledge and advice about the existing art in their fields of interest.
- 5) The quantitative allocation of specific elements of the SUMEX-AIM resource based on a concept of mean and ceiling planned expectations.

#### FACILITY INFORMATION

The computer facility, consisting of a DEC Model KI-10 CPU running under the TENEX operating system, has 256K words (36-bit) of high-speed memory, 1.6M words of swapping storage, 70M words of disk storage, two 9-track 800 bpi industry-compatible tape units, a dual DEC-tape unit, a line printer, and communications-network interfaces providing user terminal access. SUMEX is available through TYMNET and as a host over the ARPANET communications network.

Program (software) support will evolve from the basic system as dictated by the research goals and needs of the user. Initially, available programs include a variety of TENEX user, utility and text editor programs. Major user languages include INTERLISP, SNOBOL, SAIL, FORTRAN-10, BLISS-10, BASIC, Macro-10, OMNIGRAPH and MLAB.

#### POTENTIAL USERS

For further information, write:

Elliott Levinthal, Ph.D.  
 AIM User Liaison  
 SUMEX-AIM Computer Project  
 c/o Department of Genetics, S047  
 Stanford University Medical Center  
 Stanford, California 94305

Procedures for access to SUMEX-AIM are governed by the:

Biotechnology Resources Branch  
Division of Research Resources  
National Institutes of Health  
Building 31, Room 5B19  
Bethesda, Maryland 20014

## APPENDIX I

## Detailed Questionnaire for Prospective New Users

SUMEX-AIM RESOURCE  
INFORMATION FOR POTENTIAL USERS

National users may gain access to the facility resources through an advisory panel for a national program in artificial intelligence in medicine (AIM). The AIM Advisory Group consists of members-at-large of the AI and medical communities, facility users and the Principal Investigator of SUMEX as an ex-officio member. A representative of the National Institutes of Health-Biotechnology Resources Branch (NIH-BRB) serves as Executive Secretary.

Under its enabling 5-year grant, the SUMEX-AIM resource is allocated to qualified users without fee. This, of course, entails a careful review of the merits and priorities of proposed applications. At the direction of the Advisory Group, expenses related to communications and transportation to allow specific users to visit the facility may be covered as well.

## USER QUALIFICATIONS

In general terms, potential users of the SUMEX-AIM facility are judged on the basis of:

- 1) The scientific interest and merit of the proposed research.
- 2) The relevance of the research to the artificial intelligence approach of SUMEX-AIM as opposed to other computing alternatives.
- 3) The user's prospective contributions and role in the community, e.g., developing and sharing new systems or applications programs, sharing use of special hardware, etc.
- 4) The user's capability and intentions of operating in a community-effective style for mutual advantage. Besides the programming innovations that some participants may contribute, all are expected to furnish expert knowledge and advice about the existing art in their fields of interest.
- 5) The quantitative allocation of specific elements of the SUMEX-AIM resource based on a concept of mean and ceiling planned expectations.

In many respects, this requires a different kind of information for judgment of proposals than that required for routine grant applications seeking monetary funding support. Information furnished by users also is indispensable to the SUMEX staff in conducting their planning, reporting and operational functions.

The following questionnaire encompasses the main issues concerning the Advisory Group. However, this should neither obstruct clear and imaginative presentation nor restrict format of the application. The potential user should prepare a statement in his own words using previously published material or other documents where applicable. In this respect, the questionnaire may be most useful as a checklist and reference for finding in other documentation the most cogent replies to the questions raised.

For users mounting complex and especially nonstandard systems, the decision to affiliate with SUMEX may entail a heavy investment that would be at risk if the arrangement were suddenly terminated. The Advisory Group endeavors to follow a responsible and sensitive policy along these lines--one reason for cautious deliberation; and even in the harshest contingencies, it will make every effort to facilitate graceful entry and departure of qualified users. Conversely, it must have credible information about thoughtful plans for long-term requirements including eventual alternatives to SUMEX-AIM. SUMEX-AIM is a research resource, not an operational vehicle for health care. Many programs are expected to be investigated, developed and demonstrated on SUMEX-AIM with spinoffs for practical implementation on other systems. In some cases, the size, scope and probable validation of clinical trials would preclude their being undertaken on SUMEX-AIM as now constituted. Please be as explicit as possible in your plans for such outcomes.

Applicants, therefore, should submit:

- 1) One to two-page outline of the proposal.
- 2) Response to questionnaire, cross-referenced to supporting documents where applicable.
- 3) Supporting documents.
- 4) List of submitted materials, cross-referenced.

We would welcome a draft (2 copies) of your submission for informal comment if you so desire. However, for formal consideration by the SUMEX-AIM Advisory Group, please submit 13 copies of the material requested above in final form.

Elliott Levinthal, Ph.D.  
AIM User Liaison  
SUMEX-AIM Computer Project  
c/o Department of Genetics, S047  
Stanford University Medical Center  
Stanford, California 94305

May, 1975



## SUMEX-AIM RESOURCE

QUESTIONNAIRE  
FOR POTENTIAL USERS

Please provide either a brief reply to the following or cite supporting documents.

## A) MEDICAL AND COMPUTER SCIENCE GOALS

- 1) Describe the proposed research to be undertaken on the SUMEX-AIM resource.
- 2) How is this research presently supported? Please identify application and award statements in which the contingency of SUMEX-AIM availability is indicated. What is the current status of any application for grant support of related research by any federal agency? Please note if you have received notification of any disapproval or approval, pending funding, within the past three years. Budgetary information should be furnished where it concerns operating costs and personnel for computing support. Please furnish any contextual information concerning previous evaluation of your research plans by other scientific review groups.
- 3) What is the relevance of your research to the AI approach of SUMEX-AIM as opposed to other computing alternatives?

## B) COLLABORATIVE COMMUNITY BUILDING

- 1) Will the programs designed in your research efforts have some possible general application to problems analogous to that research?
- 2) What application programs already publically available can you use in your research? Are these available on SUMEX-AIM or elsewhere?
- 3) What opportunities or difficulties do you anticipate with regard to making available your programs to other collaborators within a reasonable interval of publication of your work?
- 4) Are you interested in discussing with the SUMEX staff possible ways in which other artificial-intelligence research capabilities might interrelate with your work?
- 5) If approved as a user, would you advise us regarding collaborative opportunities similar to yours with other investigators in your field?

## C) HARDWARE AND SOFTWARE REQUIREMENTS

- 1) What computer facilities are you now using in connection with your research or do you have available at your institution? In what respect do these not meet your research requirements?
- 2) What languages do you either use or wish to use? Will your research require the addition of major system programs or languages to the system? Will you maintain them? If you are committed to systems not now maintained at SUMEX, what effort would be required for conversion to and maintenance on the PDP-10 - TENEX system? What are the merits of the alternative plan of converting your application programs to one of the already available standards? Would the latter facilitate the objectives of Part B), Collaborative Community Building?
- 3) Can you estimate your requirements for CPU utilization and disk space? What time of day will your CPU utilization occur? Would it be convenient or possible for you to use the system during off-peak periods? Please indicate (as best you can) the basis for these estimates and the consequences of various levels of restriction or relaxation of access to different resources. SUMEX-AIM's tangible resources can be measured in terms of:
  - a) CPU cycles.
  - b) Connect time and communications.
  - c) User terminals (In special cases these may be supported by SUMEX-AIM.).
  - d) Disk space.
  - e) Off-line media-printer outputs, tapes (At most, limited quantities to be mailed.).

Can you estimate your requirements? With respect to a) and b), there are loading problems during the daily cycle.--Can you indicate the relative utility of prime-time (0900-1600 PST) vs. off-peak access?

- 4) What are your communication plans (TYMNET, ARPANET, other)? How will your communication and terminal costs be met? See following note concerning network connections to SUMEX-AIM.
- 5) If this is a development project, please indicate your long-term plans for software implementation in an applied context keeping in mind the research mission of SUMEX-AIM.

Our procedures are still evolving, and we welcome your suggestions about this framework for exchanging information. Needless to say, each question should be qualified a) "insofar as relevant to your proposal", and b) "to the extent of available information".

Please do not force a reply to a question that seems inappropriate. We prefer that you label it as such so that it can be dealt with properly in future dialogue.

Above all, we are eager to work with potential users in any way that would help minimize bureaucratic burdens and still permit a responsible regard for our accountability both to the NIH and the public. Please do not hesitate to address the substance of these requirements in the format most applicable to you.

#### NETWORK CONNECTIONS TO SUMEX-AIM

##### TYMNET

Attached is a list of available TYMNET nodes and associated telephone numbers. The cost to users of using TYMNET is the telephone charge from user location to the nearest TYMNET node. This is available only for communication to SUMEX-AIM and not for other facilities that may be connected to TYMNET. In some cases, there are "foreign exchanges" set up by users. These may offer less expensive communication. Details of these possibilities can best be learned by calling the nearest TYMNET node. The telephone company can provide information on comparative costs of leased lines, toll charges, etc. The initial capital investment for TYMNET installation as well as login and hourly charges is provided by SUMEX-AIM. Standard usage charges on TYMNET are approximately \$3/connect-hour.

##### ARPANET

SUMEX-AIM is connected to the ARPANET. Our name is SUMEX-AIM; our nickname is AIM. We support the new TELNET protocol. Our network address is decimal 56, octal 70. This provides convenient access for ARPANET Hosts and Associates and those who have accounts with ARPANET.

Attachment: Network service node access for TYMCOM-III users

May, 1975

## APPENDIX J

## Response to Congressional Inquiry

The following is in response to a congressional inquiry to NIH-BRB about aspects of the SUMEX-AIM resource. The questions posed include:

- 1) How much of the SUMEX resource is funded by NIH-BRB?
- 2) How many units (projects and individuals) are supported by the resource?
- 3) What is the cost per unit in operating the resource?

- 1) The SUMEX-AIM resource is essentially wholly funded by NIH-BRB.[\*] The various collaborator projects which use SUMEX are independently funded with respect to their manpower and operating expenses. They obtain from SUMEX, without charge, access to the computing and, in most cases, communications facilities in exchange for their participation in the scientific and community building goals of SUMEX.

[\*] Except for the participation by Stanford University in accordance with general cost-sharing, and for assistance to SUMEX by other projects with overlapping aims and interests.

- 2) The available SUMEX-AIM resource capacity is allocated to a variety of projects engaged in advanced computer science research (artificial intelligence) applied to medical problems. These are divided into three main groups: the projects local to the Stanford medical community (40% of the resource), the projects representing research efforts at other centers around the country (40%), and the resource development and operations staff (20%). The following gives a brief summary of the complement of projects with approximate size indicated by the total number of project members with access privileges to SUMEX and the number who were active during the latest statistics period of March. The list includes the current group of projects and may be expected to expand by 50-100% over the next year before the resource capacity is consumed. Note that each project, and in many cases each named user-member may in fact represent the efforts of from 1 to perhaps as many as 5 people sharing the same account.

## SUMEX-AIM (National Group of Projects)

=====

	Total Members	Active Members
i) AIM Community Management and Committees[*]		
Full members	8	3
Staff	5	4
ii) DIALOG Project		
Prof. Pople		
Univ. of Pittsburgh	3	3
iii) Distributed Data Bases		
Prof. Kuo		
Univ. of Hawaii	2	1
iv) Higher Mental Functions		
Prof. Colby		
Univ. Calif. at LA	4	1
v) Medical Information Systems		
Prof. McCormick		
Univ. of Illinois	8	3
vi) Computers in Biomedicine		
Prof. Amarel		
Rutgers University		
Local users	27	24
Remote users	11	4
	----	----
TOTALS	68	43

[\*] There are several additional committee members representing the user community. They are not counted under AIM management, but rather under their appropriate user project heading.

## SUMEX-SUMC (Stanford Group of Projects)

=====

	Total Members	Active Members
i) DENDRAL Project		
Prof. Djerassi, Feigen-		
baum, and Lederberg		
Local users	31	21
Remote users	9	6
ii) Information Proc Psych.		
Prof. H. Cohen and		
Feigenbaum	2	1
iii) MYCIN Project		
Prof. S. Cohen and Dr.		
Buchanan		
Local users	10	5
Remote users	4	2
iv) Protein Structure Modeling		
Prof. Freer (Univ. Calif.		
at SD) and Dr. Engelmores		
(Stanford)	4	4
v) Pilot Projects	13	6
	----	----
TOTALS	73	45

## SUMEX-SYSTEM AND STAFF (Development and Operations)

=====

	Total Members	Active Members
i) Develop. and Opns		
Staff	30	22
	----	----
TOTALS	30	22
	=====	=====
GRAND TOTALS	171	110

- 3) The cost per unit is difficult to state in a meaningful way as the usage load varies from project to project and from individual to individual. The following are gross averages computed from the estimated budget of the SUMEX-AIM project over the 5 year period of the grant award and the current user project load. Note that the project is still young and growing with respect to the user community and one may expect the number of users to increase by 50-100% over the next year without an increase in estimated project costs. This reflects the fact that the computer is not completely loaded with the present complement of users.

Project budget (5 years) [\*]

Equipment purchase	\$1,000,000
Operating budget (manpower, supplies, etc.)	\$2,245,000
	-----
	\$3,245,000

[\*] Note these funds include approximately \$100,000 per year allocated in reserve for communication usage and inter-project collaborative linkages.

This total figure is equivalent to \$649,000 per year, uniformly spreading the project costs over 5 years.

If we do not count the resource development and operations usage (which may be considered overhead in terms of medical applications), the total number of projects currently on the system is 11 (6 from the national community and 5 from Stanford). Furthermore, counting only the active users as of the March data, we have 88 people accessing the resource for research computing related to AI in medicine.

Thus, the annual costs per unit based on the current initial loading are:

\$59,000 per project per year

\$7,375 per user per year

As the user community grows by 50-100%, these unit costs will drop by 30-50%.

It should be stressed that the SUMEX-AIM management is actively involved at the present time in identifying and evaluating several additional significant projects, and that we believe that the number of projects will be substantially augmented during the next 12 months. However, the existing projects having been defined as major, on-going sites of significant research in this field are expected to continue to play a leading role in the usage statistics.

Besides the primary research activities and the corresponding users, projects like Stanford's "DENDRAL" and "MYCIN" and Rutgers's "Computers in Biomedicine" are actively dedicated to involving remote users scattered throughout the country, and using the data network facilities for the coordination of research.



**\*\*\* ADDED NOTE OF EXPLANATION \*\*\***

The people at SUMEX are happy to display the figures just given in the terms requested. However, they believe that this calculation, though obviously useful for one form of managerial perspective, may neglect some special aspects of the SUMEX-AIM experiment. Considerable effort is being devoted by the staff at SUMEX to the technical and managerial tasks of making complex computer programs more readily available and useful to a wide range of prospective users which ultimately will exceed those who are actually connected directly to the system. The SUMEX-AIM system from the start has been founded on the idea that it was necessary to build a new kind of community effort so that workers at distant sites would be able to cooperate efficiently in the solution of very complex problems. To do this requires a great deal of dedication, effort and imagination in the service of others, which is not measured by the numbers of dollars spent per active user at a given time, but by the eventual cumulative value of this pattern of research. The cost of computer access per user is believed to be quite reasonable; and every effort is made to temper this in relation to the highest-priority needs of the community it serves. However, the design of SUMEX was not founded on the idea of producing the maximum number of computer cycles per dollar or per investigator, regardless of the results that these achieve, but rather to demonstrate a cooperative mode of resource-sharing that would generate the most creative research outcomes from the aggregate efforts of its workers.

Already, its users have given many testimonials to the much enhanced efficiency with which they can pursue their work in computer science applications in medical research as a consequence of this way of working in close inter-communication.